



Universidad
Carlos III de Madrid



Departamento de Ingeniería de Sistemas y Automática

PROYECTO FIN DE CARRERA

TELEOPERACIÓN DE UN ROBOT TERRESTRE MEDIANTE BIOPOTENCIALES

PROYECTO DE COLABORACIÓN:

National Instruments Spain - Starlab

Autor: Dimas Gimeno Calvo

Tutores: Alberto Jardón Huete

Cesar Verdejo

Agradecimientos

A mis padres que siempre han sido un ejemplo de esfuerzo y un apoyo siempre que han podido. A mis hermanos, porque cada uno de ellos me aporta diferentes puntos de vista de los que intento aprender y que me enriquecen, aunque sin perspectiva es difícil saber qué te aporta alguien que tienes a tu lado, no necesito saber qué es para saber que está ahí.

También a esos otros hermanos, porque si soy lo que soy es porque nos hemos construido a base de experiencias y ellos siempre han sido un apoyo y un desahogo en momentos de tensión y de alegría. No quiero dar nombres, pero todos sabéis quienes sois.

No puedo evitar acordarme tampoco de esos otros padres que he tenido y sin los que probablemente no habría terminado estos estudios. Porque fuisteis un grandísimo apoyo en momentos difíciles y he aprendido cosas muy importantes de vosotros.

A mis compañeros de fatigas dentro y ahora también fuera de la universidad, porque no hay cosa que más una que el compartir sufrimientos y ayudarnos unos a otros a superarlos. Sé que sin vosotros no habría sido capaz de sacar todo el partido que le he sacado a mis años de universidad.

A mis compañeros de trabajo con los que he aprendido muchas cosas y siempre han supuesto un gran apoyo moral y una gran fuente de ideas cuando el proyecto parecía que no iba a llegar a buen puerto.

A esos profesores que se preparan las clases y se entusiasman con la enseñanza, porque son capaces de transmitir sus conocimientos. Y también a esos que no lo hacen, porque nos hacen ver el esfuerzo de los primeros.

Resumen

Este es un proyecto que se ha realizado en colaboración con National Instruments y Starlab. El objetivo del proyecto es la integración del hardware Enobio de la empresa Starlab y LabVIEW, entorno de desarrollo de la empresa National Instruments.

Para comprobar el buen funcionamiento y la compatibilidad se va a crear una demo en la que se van a adquirir los biopotenciales mediante Enobio y se va a realizar un filtrado de las señales y a diseñar una estrategia de control de un robot DaNI en LabVIEW. Para conseguir esto se utilizará el programa DIAdem de National Instruments como apoyo para el estudio de las señales y la toma de decisiones. Además de esto se van a añadir funcionalidades como la posibilidad de controlar el robot utilizando el teclado o ratón o un mando de la consola Wii.

Para poder realizar todo esto se deberá crear una capa de compatibilidad entre las bibliotecas de acceso a Enobio, proporcionadas por Starlab, y LabVIEW ya que no son compatibles. Después habrá que filtrar las señales adquiridas y establecer una estrategia de control del robot.

Para los demás controles se han añadido otras funcionalidades según las posibilidades de cada uno.

También se añadirá al proyecto la posibilidad de controlar una simulación 3D del robot DaNI en el entorno de desarrollo LabVIEW, de forma que esto acelerará enormemente la ejecución del proyecto, pudiendo de esta manera disminuir los tiempos estimados.

Abstract

This is a project that has been done in collaboration with National Instruments and Starlab. The project objective is the integration of hardware Enobio from Starlab company and LabVIEW development environment from National Instruments company.

To test the functionality and compatibility it is going to create a demo which acquired biopotentials from Enobio and will perform filtering of signals and create a control strategy to control a DaNI robot in LabVIEW. To achieve this, the program will use DIAdem from National Instruments as support for the study of signals and decision making. Besides this it will add features like the ability to control the robot using the keyboard or mouse or Wii Remote.

To perform this it must create a compatibility layer between Enobio access libraries, provided by Starlab, and LabVIEW as they are not compatible. After acquired, the signals must be filtered and set a strategy to control the robot.

For other controls have been added other features as the possibilities of each one.

Also added to the project the ability to control a 3D simulation of DaNI robot in the LabVIEW development environment, so that this will greatly speed up project implementation and can thus reduce the time estimates.

Índice

1.	INTRODUCCIÓN Y OBJETIVOS	1
1.1.	INTRODUCCIÓN Y MOTIVACIÓN	1
1.2.	OBJETIVOS	2
1.3.	FASES DEL DESARROLLO	4
1.4.	ESTRUCTURA DE LA MEMORIA	6
2.	ESTADO DEL ARTE	9
2.1	LA BIOMETRÍA.....	9
2.1.1	LOS ELECTROGRAMAS	9
2.1.2	LOS ARTEFACTOS	11
2.2	FILTRADO Y PROCESAMIENTO DE LAS SEÑALES.....	14
2.3	EL EOG EN MAYOR PROFUNDIDAD.....	16
2.3.1	EL ORIGEN DE LA SEÑAL EOG.....	17
2.3.2	OBTENCIÓN DE LA SEÑAL EOG.....	17
2.3.3	ARTEFACTOS TÍPICOS EN LAS SEÑALES EOG	18
2.3.4	APLICACIÓN DE FILTROS A SEÑALES EOG	19
2.3.5	APLICACIONES DEL EOG	19
2.4	LA PROGRAMACIÓN GRÁFICA	20
2.4.1	LABVIEW COMO LENGUAJE DE PROGRAMACIÓN.....	21
2.5	ROBOTS DE INVESTIGACIÓN Y SENSORES	23
3.	HERRAMIENTAS Y PLATAFORMA UTILIZADA.....	25
3.1	ENOBIO.....	26
3.2	DaNI 1.0	28
3.2.1	sbRIO 9631	30
3.3	LabVIEW.....	31
3.3.1	INTRODUCCIÓN AL ENTORNO DE DESARROLLO.....	31
3.3.2	LabVIEW ROBOTICS.....	32
3.3.3	LabVIEW RT.....	33
3.3.4	LabVIEW FPGA	33
3.4	DIAdem	35

4. INTERCOMUNICACION Y PROGRAMACION	37
4.1 ENOBIO – LabVIEW	37
4.2 LabVIEW – LabVIEW Robotics.....	39
4.3 LabVIEW – Wii Remote	42
4.4 LabVIEW – DaNI	43
4.4.1 LabVIEW – LabVIEW RT.....	43
4.4.2 LabVIEW RT – LabVIEW FPGA.....	44
5. INTEGRACIÓN Y RESULTADOS EXPERIMENTALES	47
5.1 ADQUISICIÓN DE DATOS DE ENOBIO.....	47
5.1.1 ANÁLISIS DE DATOS CON DIAdem	51
5.2 ESTRATEGIA DE CONTROL.....	52
5.3 LAS DIFERENTES OPCIONES DE CONTROL.....	56
5.3.1 CONTROL DE LA SIMULACIÓN	56
5.3.2 CONTROL DE DaNI	57
5.3.3 CONTROL DESDE ENOBIO	58
5.3.4 CONTROL DESDE Wii Remote.....	59
5.3.5 CONTROL DESDE TECLADO/RATÓN	60
5.4 RESULTADO Y DISCUSIÓN	61
6. CONCLUSIONES Y TRABAJOS FUTUROS.....	63
6.1 ANÁLISIS CRÍTICO Y CONCLUSIONES.....	63
6.2 TRABAJOS FUTUROS.....	64
7. PRESUPUESTOS DEL PROYECTO.....	67
Bibliografía	71
Figuras.....	74
Tablas	75
Gráficos	75
Glosario de términos.....	77

1. INTRODUCCIÓN Y OBJETIVOS

1.1. INTRODUCCIÓN Y MOTIVACIÓN

En los últimos años se ha dado un fuerte cambio en todos los entornos debido a la adopción de los ordenadores por todo tipo de personas, en cualquier disciplina de conocimiento y a todos los niveles. Esto nos capacita para la gestión de ingentes cantidades de datos en múltiples entornos y para infinidad de tareas. Este fue el detonante de la aparición de National Instruments como empresa. Focalizada a implementar ésta comunicación entre máquina y realidad, y a la gestión de toda esta información obtenida del entorno para poder hacer un uso práctico de ella.

Por otra parte, la empresa Starlab se ha propuesto como misión transformar la ciencia en tecnología con un profundo impacto positivo en la sociedad. En concreto una de sus áreas más destacadas de trabajo es la neurociencia aplicada, incluyendo la electro-fisiología (EEG y EOG), aplicaciones para conducción segura, la salud (estudios de sueño), seguridad (biometría usando señales fisiológicas) e interfaces cerebro-máquina (BCI).

Teniendo la oportunidad del aprendizaje obtenido en amplias áreas tecnológicas debido al trabajo realizado en NI como ingeniero de aplicaciones, se va a desarrollar este proyecto como colaboración entre empresas tan punteras como son National Instruments y Starlab.

Aunando el hardware Enobio (desarrollado por Starlab) y DaNI 1.0 (plataforma robótica desarrollada por National Instruments), a través de una comunicación inalámbrica que permitirá la transferencia de datos entre los sensores y el sistema de control y gestión de la información, y haciendo uso de LabVIEW (plataforma principal de programación gráfica desarrollada por National Instruments) y de varios módulos complementarios, se pretende desarrollar un sistema de control de un robot civil de tierra, cuyo comportamiento podría equipararse al control de una silla de ruedas motorizada (como la que aparece en la Figura 1) para el desplazamiento de personas tetrapléjicas o con dificultad para el movimiento.



Figura 1: Silla de ruedas motorizada[1]

1.2. OBJETIVOS

Los objetivos de este proyecto se pueden contemplar en varios niveles.

Como conjunto el objetivo global del proyecto será el desarrollo de una demo que se pueda presentar en el congreso de National Instruments Spain delante de los clientes como muestra de la potencia de desarrollo y conexión con hardware de terceros que tienen los productos de la empresa. Esto se traduce en conseguir un sistema de control de la plataforma robótica DaNI 1.0 desde el hardware Enobio proporcionado por la empresa Starlab.

Moviéndose a un nivel más detallado se puede dividir en tres sistemas que propondrán diferentes retos.

Tomando como primer subsistema el hardware Enobio, se encuentra un hardware al que se tendrá que conseguir acceder a partir de las bibliotecas que proporciona el fabricante para ser capaz de conectarse, configurarlo, y acceder a los datos que proporcione para poder trabajar con ellos.

Tomando como segundo subsistema el PC, se encuentra la necesidad de desarrollar un programa que después de recibir los datos sea capaz de gestionar la información adquirida y transformarla en órdenes hacia el robot. Se utilizará en este paso el programa DIAdem y la plataforma LabVIEW Robotics junto con otras bibliotecas para el desarrollo de mayores funcionalidades como el control del robot con otros dispositivos.

Si el tercer subsistema es DaNI, aquí se encuentra la necesidad de implementar algún tipo de conexión inalámbrica externa ya que DaNI no posee esta capacidad. También se deberá desarrollar al menos parte del código de funcionamiento para la comunicación con el ordenador.

Si se concretan los objetivos en puntos más específicos, se pueden plantear los siguientes:

- Acceso a la biblioteca proporcionada por la empresa Starlab para la adquisición y manejo de toda la información captada por el hardware Enobio.
- Implementación de unas librerías o drivers (VIs y subVIs) en el entorno de programación LabVIEW para el acceso y manejo de la información obtenida del hardware Enobio.
- Programación de la FPGA para el acceso eficiente y en tiempo real a los puertos de entrada y salida de la tarjeta sbRIO 9631 que constituye el núcleo principal del sistema robótico.
- Programación de movimientos básicos del robot de tierra DaNI.

- Acceso y calibración al sensor de ultrasonidos incorporado en el robot DaNI, para una posterior utilización de la información obtenida como realimentación en el comportamiento en funcionamiento del robot (sistema anti colisión).
- Comprobación del sistema anti colisión utilizando un algoritmo de navegación autónomo.
- Programación de comunicación con el mando Wii desde una plataforma PC con bluetooth, acceso a los sensores y manejo de eventos mediante la lectura de estos.
- Programación del manejo del robot mediante el uso del mando Wii Remote para la comprobación de la programación del robot DaNI. Comprobación de movimientos sencillos como marcha adelante, marcha atrás, giro izquierda, giro derecha, y combinaciones de ellos. Comprobación del sistema anti-colisión cuando el modo de funcionamiento es manual.
- Programación del manejo del robot DaNI mediante el hardware Enobio. Según la información que sea posible obtener del sensor y la precisión de la misma será posible una programación para obtener una mayor o menor funcionalidad del robot.

Previsiblemente, se espera poder ejecutar movimientos básicos del robot utilizando las señales EOG, y captando mediante las señales EEG el estado de la persona que lo maneja, pudiendo así, por ejemplo, desconectar los motores o encender una alarma si se detecta que el sujeto se duerme.

Antes de empezar la ejecución del proyecto las personas responsables de la empresa Starlab han comunicado unos tiempos de ejecución del proyecto que han estimado siendo conocedores del hardware y la API que han desarrollado.

Su estimación la dividen en tres bloques. Un primer bloque que consistiría en la implementación del software de Enobio en LabVIEW, lo que se traduce en integrar las dlls de la API y comunicar LabVIEW con el hardware, encender, apagar, guardar ficheros de datos y mostrar las señales de los distintos canales. Esta primer parte la estiman en 2 o 3 meses trabajando una persona. Como segunda parte consideran la integración del robot en LabVIEW, lo que estiman que podría durar entorno a 1 mes. Y por último la extracción de características y control del robot, que consistirá en desarrollar los algoritmos de control del robot que pueden combinar la inteligencia del robot con los sensores. Esta última parte se estima en torno a unos 3 meses, aunque se puede alargar dependiendo de la demo final a mostrar.

1.3. FASES DEL DESARROLLO

Para el desarrollo del proyecto se ha seguido el siguiente proceso, que puede verse representado en el diagrama de Gantt de la Figura 2:

- Realización de cursos sobre el entorno de programación LabVIEW. Se realizaron los cursos de la empresa National Instruments: LabVIEW Core 1, LabVIEW Core 2, LabVIEW Real-Time, LabVIEW FPGA.
- Formación en el software DIAdem, que ha sido utilizado en el desarrollo del proyecto.
- Toma de confianza con la plataforma robótica DaNI, los diferentes aspectos de su programación y del hardware que lo compone. Dando especial atención a la placa sbRIO 9631.
- Puesta en funcionamiento de la plataforma DaNI en el entorno de simulación LabVIEW Simulator.
- Configuración para el manejo manual desde el teclado/ratón de la plataforma simulada DaNI.
- Instalación de librerías abiertas para conexión por Bluetooth con un mando de la consola Nintendo Wii. Y desarrollo de una solución para el control de la plataforma simulada DaNI mediante los datos adquiridos del mando.
- Puesta en funcionamiento de la plataforma real DaNI. Instalación de software, configuración y programación para la interacción con el software desarrollado para la plataforma simulada.
- Estudio de la API proporcionada por Starlab para el funcionamiento de Enobio. Estudio de ejemplos de programación en C++ de la API para el uso de las bibliotecas compiladas por Starlab.
- Estudio de la documentación específica "*Using external code in LabVIEW*" proporcionada por National Instruments.
- Desarrollo de un wrapper para las bibliotecas otorgadas por Starlab para permitir el acceso y comunicación entre LabVIEW y Enobio.
- Desarrollo de Vis (instrumentos virtuales), para el acceso, configuración, y comunicación con Enobio.
- Realización de un estudio sobre los datos adquiridos por Enobio mediante el software DIAdem, para establecer la mejor técnica de interpretación de la información obtenida.
- Desarrollo de varios programas para la utilización de Enobio. Programa de test de señales, entrenamiento, y finalmente programa de control para la interacción con DaNI.
- Integración del proyecto completo con todas las funcionalidades alcanzadas.
- Puesta a punto del proyecto para poder ser mostrado en un congreso con clientes de la empresa National Instruments.
- Traducción del proyecto y creación de guías explicativas en inglés para el uso del proyecto en un congreso en Austria.

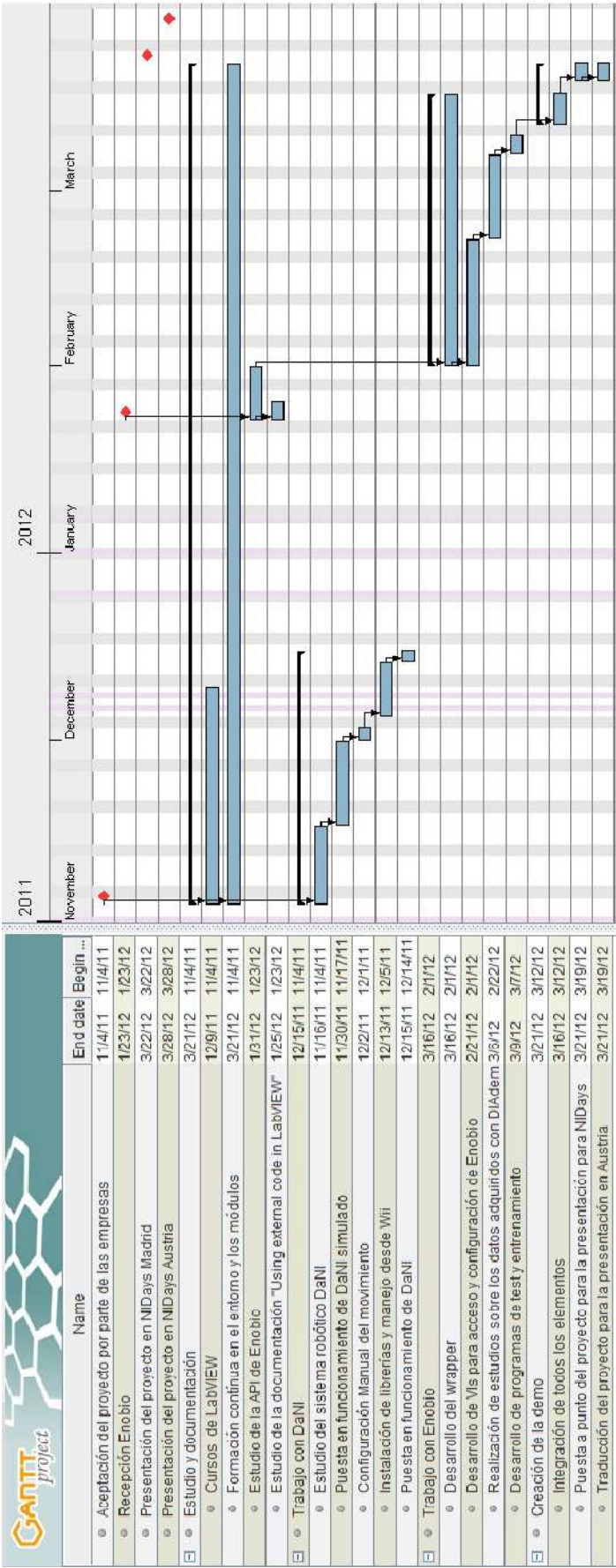


Figura 2: Diagrama de Gantt

1.4. ESTRUCTURA DE LA MEMORIA

Para facilitar la lectura de la memoria, se incluye a continuación un breve resumen de cada capítulo:

El capítulo 1 consta de una representación abreviada, objetiva y precisa del contenido del proyecto. Se presentan los objetivos que se persiguen así como las fases de desarrollo del proyecto.

En el capítulo 2 se realiza una descripción del estado de la técnica en los campos que afectan al proyecto. Comienza con una breve introducción a los sistemas de interacción hombre máquina, se continúa con una descripción de los BCI (*Brain Computer Interface*), mencionando algún dato de los más conocidos, y parecidos al utilizado en el proyecto. Finalmente se hace mención a la plataforma robótica, otra parte del hardware que constituirá parte de este proyecto final de carrera, así como una breve mención al software que se utilizará.

El capítulo 3 se centrará en la descripción detallada del software y hardware utilizado, otorgando detalles constructivos, versiones de software, y capacidades de hardware de los elementos que formarán parte del sistema final desarrollado.

El capítulo 4, siguiendo con las fases de desarrollo del proyecto se detiene en explicar la interconexión entre cada una de las partes hardware y software que van a formar el conjunto de la demo.

El capítulo 5 muestra los resultados obtenidos a lo largo del proyecto, tanto en la programación, como en la configuración hardware, y la puesta en marcha del sistema completo.

El capítulo 6 plantea las conclusiones obtenidas durante todo el proceso, y la discusión de trabajos futuros.

En el capítulo 7 se puede encontrar un análisis económico del coste estimado del proyecto si fuese realizado en el ámbito empresarial.

Terminado el último capítulo se puede obtener un listado de la bibliografía consultada, en los diferentes medios empleados.

Tras la bibliografía se pueden encontrar unos índices de figuras, tablas, gráficos y un glosario con algunos términos con los que el lector pueda no estar muy familiarizado.

Por último se encuentran los anexos que contienen documentación a la que se ha hecho referencia, o que puede ser de ayuda consultar durante la lectura del documento.

2. ESTADO DEL ARTE

2.1 LA BIOMETRÍA

Presentado de esta forma el proyecto, no se puede evitar ahora hacer una pequeña introducción a todo lo que la biometría implica en estos días. Y cuando se habla de biometría se quiere hacer en un término general, no se habla únicamente de “biometría informática” como denominan algunos autores [2], a esa parte de la biometría especializada en la medida de variables para verificar o identificar individuos, sino que se habla de las medidas de rasgos conductuales o físicos que incluyen, además de huellas dactilares, iris, etc., otros como tensión arterial, temperatura corporal, ritmo cardíaco o conductividad de la piel.

2.1.1 LOS ELECTROGRAMAS

Dentro de todo ese otro cúmulo de señales que pueden otorgar información sobre un individuo, se van a comentar las que comparten grandes similitudes, o métodos de adquisición con la que será la protagonista para el desarrollo de este proyecto, la señal EOG, electro-óculo grama.

Desde que se desarrollan electrodos Ag/AgCl, se han conseguido obtener unos sensores que son capaces de obtener medidas de tensión en la piel de un sujeto, de un alto rango en frecuencias, típicamente este tipo de sensores pueden ser usados para medir señales desde 0 a 10 KHz, salvando el problema que existía con los electrodos de Au (que no pueden usarse para la medida de tensiones continuas), y que además están sujetos a menor polarización y deriva (*drift*) [3]. En la Figura 3 puede verse un mapa de la colocación típica de esta clase de electrodos en una persona para la realización de diferentes tipos de estudios.

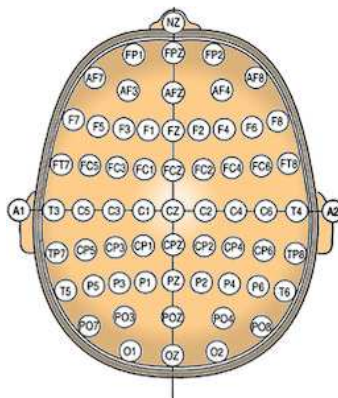


Figura 3: Posiciones de los electrodos para estudios EEG en personas[4]

TELEOPERACIÓN DE UN ROBOT TERRESTRE MEDIANTE BIOPOTENCIALES

Gracias al desarrollo de estos sensores, se ha convertido en una tarea relativamente sencilla el desarrollo de sistemas de adquisición de señales basadas en biopotenciales (tensión medida en la piel de un individuo). Este tipo de señales son más concretamente: ECG (electrocardiograma), EOG (electrooculograma), EMG (electromiografía) y EEG (electroencefalograma). También conocidas en conjunto como electrogramas[5].

Con estas señales, se es capaz de medir el pulso, movimiento de los ojos, acciones de los músculos, e incluso actividad cerebral. Y el tratamiento de toda esta información está conduciendo a proyectos tan espectaculares que seguramente hace pocos años se pensaría que eran ciencia ficción. Se está investigando y mejorando cada vez más los que se denominan BCI (*brain computer interface*) o las prótesis mioeléctricas como las desarrolladas últimamente por Ottobock [6].

Las señales que se consiguen adquirir con este tipo de sensores, se han caracterizado y se las puede describir según los valores que se pueden ver en la Tabla 1 para cada una de ellas [7]. Aunque según la fuente consultada estos valores no son siempre los mismos, algunos autores consideran distintos rangos de frecuencias para algunas de ellas.

SEÑAL	MAGNITUD	ANCHO DE BANDA (Hz)
ECG (electrocardiograma)	0.5 – 4 mV	0.01 - 250
EOG (electrooculograma)	50 – 3500 μ V	DC - 50
EMG (electromiograma)	0.1 – 5 mV	DC - 10000
EEG (electroencefalograma)	5 – 300 μ V	DC - 150

Tabla 1: Magnitudes de los electrogramas [7]

Pero no sólo se pueden caracterizar por estos valores. Las señales según sea su procedencia pueden poseer otras características como una estructura de ondas que se repiten, como es el caso del ECG, cuya onda sigue un patrón muy específico, que puede observarse en el Gráfico 1, debido a las descargas eléctricas que se producen en el corazón para provocar las contracciones del mismo.

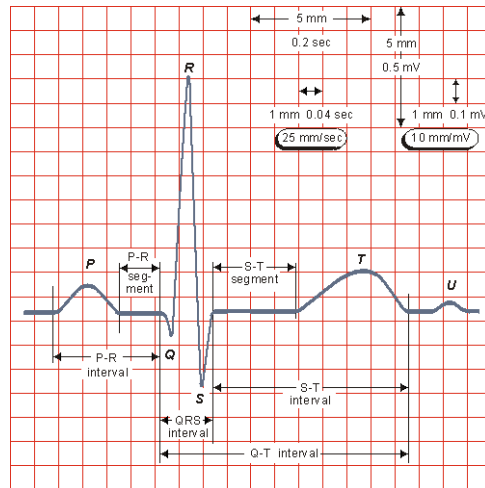


Gráfico 1: Esquema de la señal ECG [8]

Otro ejemplo es el caso del EEG, cuyas medidas se dividen para su estudio en bandas de frecuencias, ondas delta (lentas) de menos de 4 Hz; bandas theta de 4-8 Hz, las alfa de 8 a 12 Hz, las beta de aproximadamente 14-30 Hz y las gamma por encima de 30 Hz, y en algunos casos de estudio se han considerado hasta 250 Hz, por encima de lo indicado anteriormente, en investigaciones de ERP (potenciales evocados), que intentan detectar actividad EEG intracraneal [9]. Y según el estado del individuo, se hacen más acusadas cierto tipo de ondas u otras. Por ejemplo el ritmo de alfa se aprecia mejor cuando el sujeto está despierto y relajado con los ojos cerrados.

2.1.2 LOS ARTEFACTOS

Todos estos hechos llevan por otra parte a que este conjunto de señales comparten entre ellas rangos de medida en tensión y frecuencia, y como no se pueden aislar las fuentes de estas señales, esto va a provocar interferencias, o artefactos como se denominan en éste hábito (y así se les hará referencia si es necesario más adelante), entre los distintos tipos de señales en las medidas tomadas. Y en concreto en la zona de medida de interés para este proyecto, que es el lóbulo frontal, intervienen todos los electrogramas más que en ninguna otra zona.

Dentro de los artefactos en este tipo de medidas se pueden distinguir dos tipos. Un primer tipo de artefactos que se pueden denominar internos, cuya procedencia es el propio individuo objeto de estudio. Y el segundo tipo de artefactos que consideraremos serán los que afecten al sistema de medida pero tengan un origen externo al individuo de estudio.

Estos artefactos internos son esas interacciones que se ha comentado que existen entre las diferentes señales biométricas.

TELEOPERACIÓN DE UN ROBOT TERRESTRE MEDIANTE BIOPOTENCIALES

Normalmente un estudio o medida sólo va a estar interesado en obtener información de uno de los electrogramas, y por lo tanto toda la información introducida por el resto en la medida va a ser considerada ruido. Existen artefactos comunes a todos los electrogramas como son por ejemplo el estado de ánimo del individuo o la sudoración, y éstos serán considerados como artefactos sea cual sea el estudio que se esté realizando. Sin embargo existen otros artefactos propios de cada electrograma, o que afectan más a algunos que a otros como pueden ser por ejemplo cambios en la luminosidad del entorno que afectan más a los EOG y EEG que a ECG o EMG, el nivel de concentración del individuo, la respiración, e incluso los movimientos oculares o la tensión arterial y los latidos serán considerados artefactos siempre que no sean el objeto del estudio que se está realizando.

Por otra parte, entre los artefactos externos los más destacables y comunes son, el movimiento del interfaz electrodo-piel; interferencias externas al equipo de medida, capacitivas e inductivas; interferencias debidas a cargas electrostáticas; e interferencias debidas al equipo de medida provocadas por el transformador de la fuente de alimentación, el rizado de la fuente, o el ruido debido a los componentes electrónicos. Todos estos pueden ser de considerables debido a las amplitudes de las señales medidas, y al espectro de frecuencia de las mismas. En la Figura 4 se puede ver un esquema de estos artefactos externos.

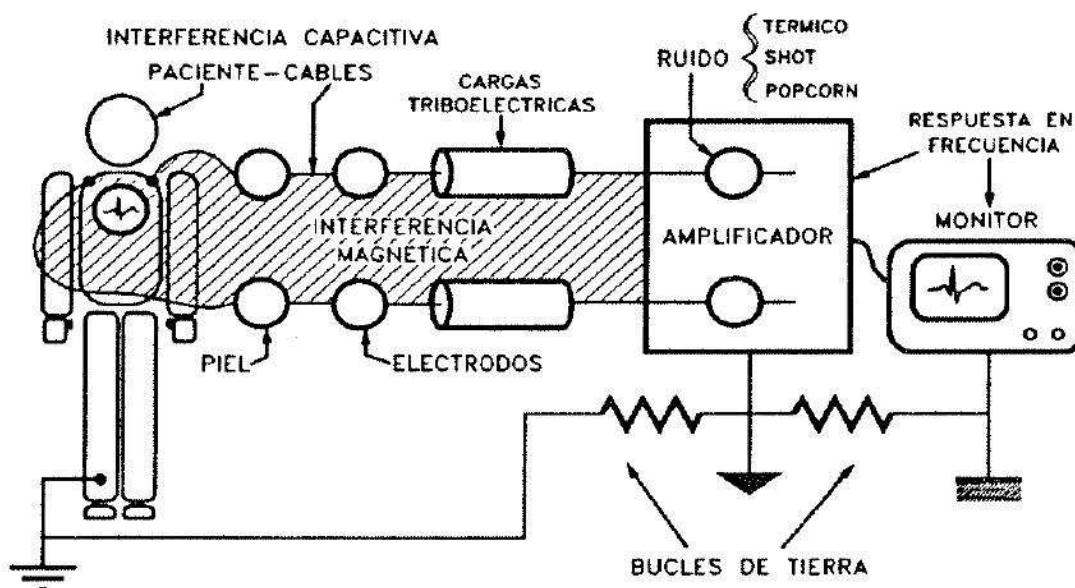


Figura 4: Interferencias externas[7]

Aunque el mayor problema de estas interferencias externas es casi siempre debido a la red de distribución de energía, que provoca una señal de interferencia a 50 o 60 Hz. Según la red eléctrica del país al que se encuentre conectado el equipo. Normalmente las interferencias introducidas por otros equipos electrónicos suele estar en frecuencias mucho más altas y son fácilmente subsanables. Y es ésta red eléctrica la responsable a su vez de las alteraciones debidas al acoplo inductivo y capacitivo.

El acoplo capacitivo se produce al conectar el sistema de medida al individuo objetivo. Se puede considerar el cuerpo humano como un conductor volumétrico. Este conductor está separado por el aire de los conductores de la red eléctrica que se encuentran en el ambiente donde se realizan las medidas. Se forman de esta manera dos condensadores cuyas armaduras son el sujeto y los conductores vivos y de tierra respectivamente, asumiendo el aire el papel de dieléctrico.

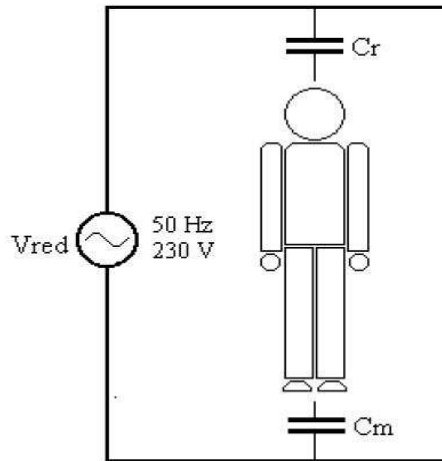


Figura 5: Acoplo capacitivo[7]

En la Figura 5 se puede ver cómo se modela este acoplo capacitivo, donde C_r representa el acoplo a la red, y en un individuo correctamente aislado alcanza un valor típico de 2 pF. Mientras que C_m que representa el acoplo con los instrumentos de medida alcanza valores típicos de 200 pF.

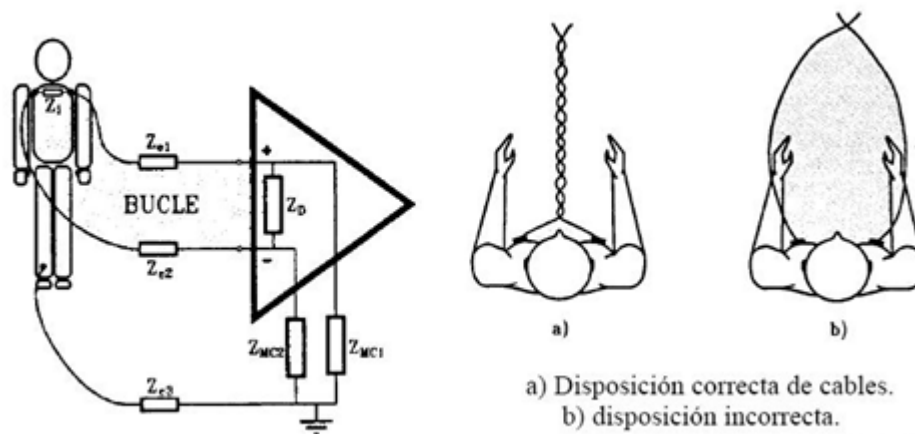


Figura 6: Bucle de interferencia y disposición de los cables [7]

Las interferencias inductivas son producidas normalmente por la red eléctrica también, ya que por ella circula una tensión alterna de una frecuencia de 50 o 60 Hz, y esto provoca unos campos magnéticos variables. Estos campos atraviesan los bucles formados por el sistema de medida induciendo en ellos tensiones a la frecuencia de la red de acuerdo con la ley de Lenz. Puede verse en la Figura 6 como evitar estas interferencias.

2.2 FILTRADO Y PROCESAMIENTO DE LAS SEÑALES

Conocidos los problemas que se pueden tener al adquirir las señales se van a plantear a continuación varios procedimientos de actuación frente a las causas que los producen.

Partiendo del último punto problemático, una solución aceptable suele ser el implementar una batería para la alimentación del sistema de amplificación de las medidas. Esto además de reducir el ruido de alimentación, añade seguridad al individuo de estudio. En cualquier caso también pueden utilizarse amplificadores aislados, o convertidores aislados junto con redes de filtrado en la alimentación para atenuar la interferencia del rizado de la fuente.

En la etapa de adquisición, los filtros utilizados son activos. Esto quiere decir que no utilizan únicamente elementos pasivos como resistencias, bobinas o condensadores para su diseño, sino que utilizan elementos activos, que proporcionan una cierta forma de amplificación de energía como tubos de vacío, transistores, o amplificadores operacionales. Para su implementación se combinan elementos pasivos y activos. Según la señal de interés pueden implementarse tanto filtros paso bajo, como paso banda, *notch* o paso alto. A continuación se hace una breve descripción de estos filtros.

Los filtros paso bajo tienen una frecuencia de corte, por encima de la cual atenúan la señal de entrada, consiguiendo de esta forma desechar las frecuencias superiores que no sean de interés en el estudio. Por ejemplo, sabiendo que las señales EOG se encuentran entre DC y 50 Hz. Se implementaría un filtro de éste tipo para rechazar el ruido introducido por la electrónica o los artefactos que introduciría el EEG en alta frecuencia.

Los filtros paso banda poseen una frecuencia de corte superior y otra de corte inferior y atenúan las frecuencias que queden por debajo de la frecuencia inferior, o por encima de la superior. De la misma forma que el filtro paso bajo, en este caso podría implementarse en la adquisición de una señal ECG, que se encuentra entre 0.01 y 250 Hz. Para eliminar la continua introducida por el mal aislamiento del individuo o por artefactos provenientes del ECG, EEG o EMG.

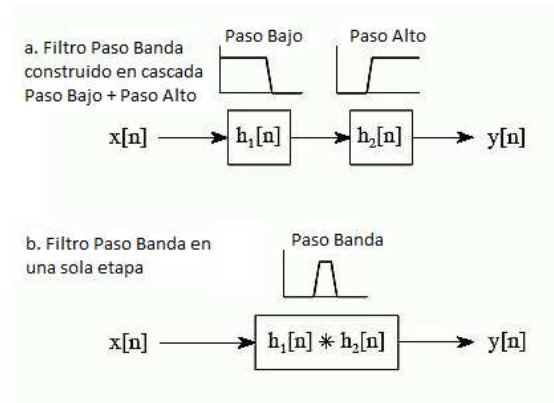


Figura 7: Filtros paso bajo, alto y banda

Los filtros paso alto, funcionan de la misma forma que los paso bajo, pero atenúan las señales con menor frecuencia que su frecuencia de corte.

Los filtros *notch*, o filtros bloqueo banda, tienen dos frecuencias de corte entre las que atenúan cualquier señal de frecuencia comprendida entre ambas. Este filtro es muy adecuado por ejemplo para eliminar los ruidos que introduce la red de alimentación en las medidas entorno a los 50 o 60 Hz. En caso de que no se haya optado por una alimentación mediante baterías DC.

Estos filtros básicos pueden verse esquematizados en la Figura 7, además pueden diseñarse de primer, segundo, o más ordenes, obteniendo pendientes mayores de atenuación. También pueden combinarse entre ellos para crear filtros más complejos.

Con un sistema de medida correctamente aislado y con los filtros de adquisición bien diseñados, la mayor preocupación será el filtrado de los artefactos internos. Para este filtrado se recurre sobre todo a dos tipos distintos. Se conocen como PCA e ICA respectivamente, y se va a comentar brevemente en qué consiste cada uno de ellos.

El PCA (*Principal Component Analysis*) o análisis de componentes principales es un método típico en el análisis de datos multivariantes cuyo objetivo es la reducción de la dimensionalidad de los mismos.

PCA es un método general de análisis de datos y se aplica en el múltiples entornos, análisis de señales biométricas; procesamiento de imágenes como reconocimiento de caras con alguna variación en el método, llamándolo *eigenfaces*; o análisis de tumores [10].

El PCA es una forma de identificar patrones en datos y representar los datos en una forma tal que sus semejanzas y diferencias sean resaltadas. Este método resulta de gran utilidad cuando es necesario buscar cierto tipo de información. Esta técnica se utiliza desde hace tiempo para reducir los artefactos entre electrogramas como EEG y EOG [11].

El ICA (*Independent Component Analysis*) o análisis de componentes independientes, consiste en transformar un conjunto de señales observadas en un conjunto de señales estadísticamente independientes llamadas fuentes o componentes independientes del primer conjunto.

El principal objetivo de ICA es identificar a partir de las señales observadas, un nuevo grupo de señales subyacentes con significado. Esta representación puede ser usada en distintos ámbitos como la extracción de características y el reconocimiento de patrones.

Esta técnica difiere de los análisis en componentes principales en que ICA impone independencia de alto orden y no solo de segundo orden como en PCA. Y ya ha sido aplicada para la reducción de artefactos en señales biométricas [12].

2.3 EL EOG EN MAYOR PROFUNDIDAD

En la actualidad existen cuatro técnicas de exploraciones electrofisiológicas del ojo humano. El electroretinograma (ERG), que registra la respuesta eléctrica de las células sensibles a la luz que se encuentran en el ojo; el videooculograma (VOG) que es capaz de detectar los movimientos del ojo humano basándose en imágenes adquiridas por una cámara; los potenciales evocados visuales (PEV) que registran cambios en la actividad cerebral después de haber recibido estímulos visuales a través del ojo; y por último el electrooculograma (EOG).

Como ya se ha comentado en puntos anteriores de este escrito, el EOG o electrooculograma es uno de los biopotenciales medibles con los sensores Ag/AgCl. Para poder entender qué es exactamente lo que se va a medir se empezará haciendo una pequeña introducción anatómica al ojo humano.

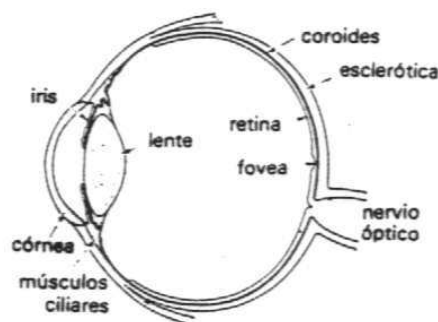


Figura 8: El ojo humano[13]

2.3.1 EL ORIGEN DE LA SEÑAL EOG

El globo ocular (Figura 8) es una estructura esférica de aproximadamente 2.5 cm de diámetro. Está formado por varias capas entre las que se encuentran la córnea, situada en la parte exterior del globo ocular y a través de la cual la luz penetra en el interior del ojo, y la retina, capa interna y compleja compuesta sobre todo por células nerviosas sensibles a la luz, con forma de conos y bastones, y dispuestas como los fósforos de una caja de cerillas.

Esta disposición de las células provoca una diferencia de potencial entre la córnea y la retina que permite considerar la presencia de un dipolo, donde la córnea corresponde al extremo positivo y la retina al extremo negativo de dicho dipolo.

2.3.2 OBTENCIÓN DE LA SEÑAL EOG

El EOG es el método de registro de los movimientos oculares basado en la diferencia de potencial existente entre la córnea y la retina.

La retina tiene un potencial bioeléctrico de reposo de carácter electronegativo respecto de la córnea. De este modo, los giros del globo ocular provocarán cambios en los valores medidos por los sensores según la dirección del vector correspondiente a este dipolo eléctrico, ver Figura 9.

Pero hay que destacar en este momento que no todos los movimientos de los ojos pueden ser medidos con igual precisión.

El ojo humano puede realizar distintos tipos de movimientos, se van a comentar a continuación algunos de ellos para entender las diferencias y el porqué los sensores responderán mejor a unos o a otros.

Los movimientos compensatorios, son movimientos reflejos que implican la participación de ambos ojos, y surgen como mecanismo de fijación del campo visual ante movimientos de la cabeza o del tronco. Los movimientos de seguimiento o persecución lenta, son movimientos que se producen de forma coordinada con ambos ojos y cuya finalidad es la de seguir estímulos visuales que se desplazan lentamente. Los movimientos de vergencia implican el movimiento de los ojos en direcciones opuestas, su finalidad es la de proyectar correctamente la imagen en las retinas para obtener una única imagen fusionada, este movimiento se lleva a cabo cuando el objeto en que se fija la vista se mueve en la dirección del observador ya sea acercándose o alejándose de él. Por último, los movimientos sacádicos, constituyen uno de los movimientos más característicos de los ojos. Son movimientos fundamentalmente voluntarios (aunque los hay involuntarios como los que tienen lugar en la fase REM de sueño). Nos permiten visualizar diversas zonas de una misma escena. En esencia su objetivo es el de disponer la imagen visual en la fóvea que es la región de la retina que

dispone de una mayor agudeza visual. Es lo que ocurre cuando en una escena vamos fijando la mirada en distintos puntos de la misma. Pero lo más destacable de éste tipo de movimientos es que durante la ocurrencia del sacádico (o sacada) se ocasiona el denominado fenómeno de supresión sacádica, es decir, durante el movimiento la recogida de información visual del medio queda prácticamente interrumpida [14]. Y es éste fenómeno también va a facilitar la medida de los sensores.

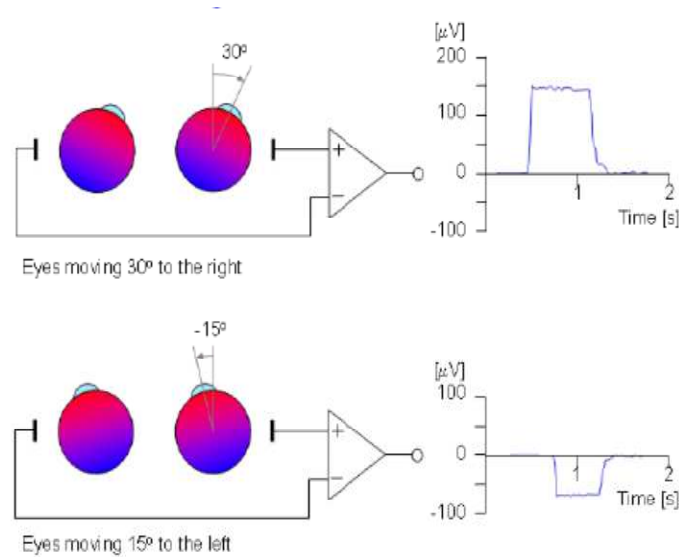


Figura 9: Registro de movimientos oculares[15]

2.3.3 ARTEFACTOS TÍPICOS EN LAS SEÑALES EOG

Como ya se ha comentado con anterioridad, los diferentes electrogramas van a provocar interferencias en las medidas de los demás ya que comparten rangos de medida en tensión y en frecuencia. En concreto, las medidas de un electrooculograma se realizan alrededor de los ojos, y esta resulta ser la zona donde más interaccionan éstas señales ya que en la misma zona pueden encontrar músculos, grandes venas y el cerebro, a parte de los ojos que son el verdadero objetivo de las medidas.

A parte de estos artefactos, en las medidas de un EOG también van a intervenir cambios en la intensidad luminosa del entorno, estado de ánimo de la persona, artefactos de movimiento del interfaz electrodo – piel, nivel de concentración al realizar los movimientos, sudoración, o cambios de temperatura.

2.3.4 APLICACIÓN DE FILTROS A SEÑALES EOG

Se han comentado con anterioridad algunos tipos de filtros que se aplican en los electrogramas. A la hora de la eliminación de artefactos en las señales EOG van a ser de interés para los siguientes tipos de filtros.

El filtro paso bajo, es la mejor opción para eliminar algunos de los artefactos que introducen en la medida la señal EEG y la ECG, que como se ha visto coexisten con el EOG. Estas alcanzan unos rangos de frecuencias superiores a los que interesa medir. Ya que el rango de frecuencias para un electrooculograma alcanza un máximo de entre 50 Hz y 100 Hz (según autores consultados), sería interesante aplicar un filtro paso bajo con frecuencia de corte situada en el máximo valor de frecuencias que se desea conservar y según las necesidades se puede aplicar de primer, segundo o incluso de órdenes más altos.

El filtro *notch* o bloquea banda, también es muy interesante para eliminar el ruido introducido por la red, y se debería diseñar con una frecuencia de corte en torno a los 50 o 60 Hz, según la frecuencia de la red del lugar en el que se tomen los datos.

Existe también otro tipo de filtrado que se ha empleado en algunos estudios, pero que no se aplica de forma masiva todavía. Se trata de la transformada compleja de wavelets o CWT [16]. No es un filtrado de aplicación común, pero probablemente la razón se a que el desarrollo de este tipo de filtrado no ha comenzado a utilizarse hasta 1990, cuando David Donoho y Johnstone utilizan por primera vez los wavelets para eliminar el ruido de una señal. El punto fuerte del análisis y filtrado mediante wavelets reside en que el análisis mediante wavelets da información sobre el espectro de frecuencias en función del tiempo. No como un análisis de Fourier, que permite determinar sus frecuencias, pero a costa de perder la información de tipo temporal sobre la señal. Sin embargo dado el ajustado calendario de ejecución del proyecto no se ha podido comprobar la aplicabilidad de este tipo de filtros en este proyecto, y se sugerirá como trabajo futuro ya que sí se ha observado un potencial importante en ellos.

2.3.5 APLICACIONES DEL EOG

El EOG no es aplicado hoy en día como un sistema de diagnóstico por sí mismo, sin embargo sí que se utiliza en conjunción con otros sistemas para la detección de diversas alteraciones como son por ejemplo:

- Distrofias del epitelio pigmentario retinal:
 - Enfermedad de Best
 - Enfermedad de Stargardt
- Diagnóstico precoz de toxicidad retiniana por medicamentos.
- Diagnóstico de enfermedades relacionadas con la alteración del sueño:

- Narcolepsia
- Síndrome de apneas obstructivas durante el sueño
- Trastorno de conducta durante el sueño REM

A parte de estas aplicaciones médicas, se han desarrollado varios sistemas de interfaz hombre-máquina a partir de la medida del EOG. Pero no se ha encontrado referencia a ningún sistema de este estilo comercializado.

2.4 LA PROGRAMACIÓN GRÁFICA

Se ha hecho ya una introducción a los métodos de adquisición de las señales que se utilizarán durante el desarrollo del proyecto, pero la adquisición es sólo una parte del mismo. Se va a continuar con una pequeña introducción al entorno de programación gráfica.

Históricamente los lenguajes de programación siempre han tenido una sintaxis textual, de ahí su denominación, constaban de elementos como variables, vectores, condicionantes, bucles y funciones. Para la creación de un programa era necesario un estudio previo del lenguaje y mucha práctica para poder desarrollar código correcto, claro, eficiente y portable en la medida de lo posible. Pero con la extensión en el uso de los ordenadores a todos los ámbitos, se hace si no necesario, si de mucha ayuda, la creación de un entorno de desarrollo fácil de utilizar por personas que no sean programadores experimentados. Nacen de esta forma los llamados lenguajes gráficos de programación, entre los que hoy en día podemos encontrar Yenka [17], Flowol [18], Profilab Expert [19] o LabVIEW [20].

Los lenguajes gráficos son mucho más intuitivos que los lenguajes de sintaxis textual. Utilizan bloques funcionales (ver Figura 10) muy potentes y suelen tener drivers para la conexión con hardware del mundo exterior al ordenador de desarrollo.

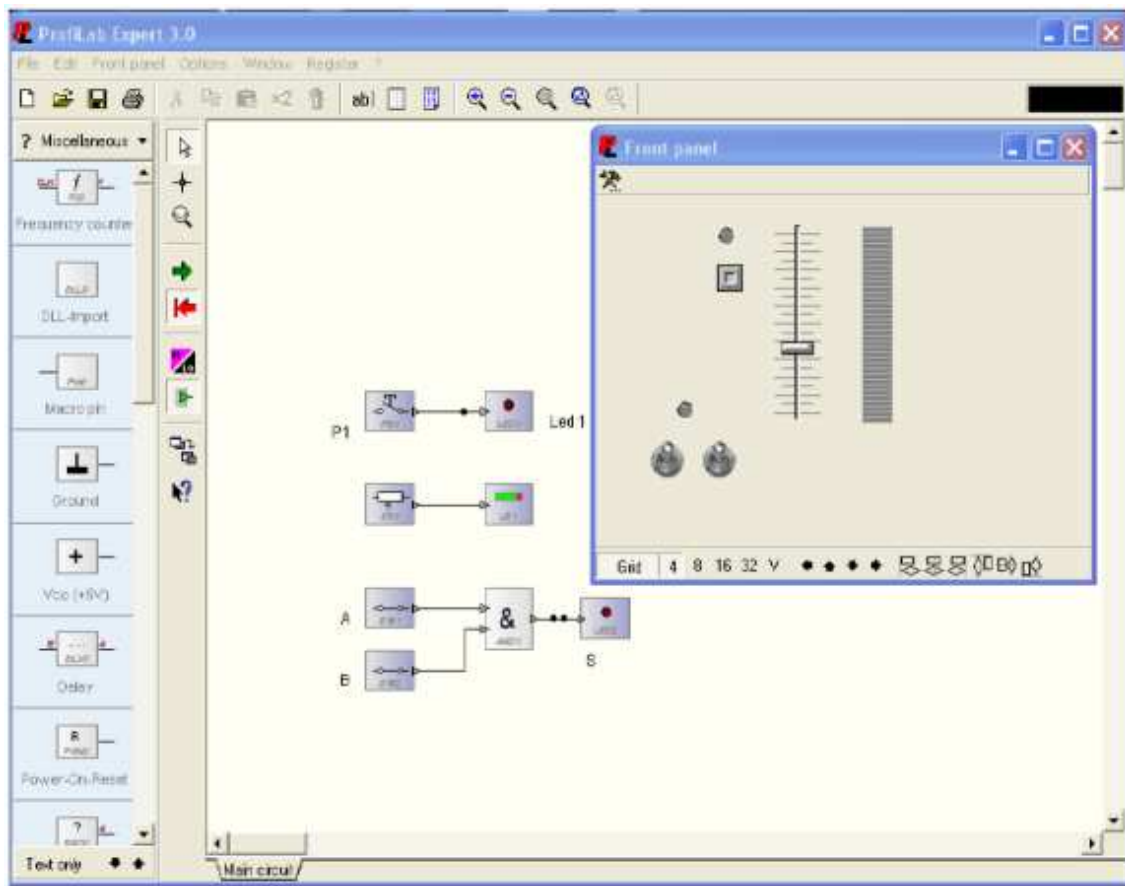


Figura 10: Ejemplo de programación gráfica en ProfiLab Expert[19]

Constan de amplias librerías con las que se pueden crear programas únicamente enlazando unas con otras hasta conseguir la funcionalidad deseada. Y ofrecen la posibilidad al usuario de encapsular partes de código desarrollado por él mismo y de esta manera crear sus propias librerías.

2.4.1 LABVIEW COMO LENGUAJE DE PROGRAMACIÓN

Dentro de los lenguajes de programación gráfica LabVIEW provee de una variada gama de características y herramientas de asistentes e interfaces de usuario configurables, se diferencia por ser un lenguaje de programación gráfico de propósito general (conocido como G), con su compilador asociado, su enlazador, y herramientas de depuración.

LabVIEW es diferente de la mayoría de lenguajes de propósito general principalmente en dos vertientes. Primero, la programación G se desarrolla cableando iconos gráficos en un diagrama que compila directamente a código máquina de modo que los procesadores del ordenador pueden ejecutarlo

El segundo diferenciador principal es que el código G desarrollado en LabVIEW se ejecuta de acuerdo con las reglas del flujo de datos en lugar del acercamiento más tradicional (en otros términos, una serie secuencial de comandos para ser llevados a cabo) que se encuentran en la mayoría de los lenguajes de programación basados en texto como C y C++. Los lenguajes de flujo de datos como G (también VEE de Agilent, Microsoft Visual y Apple Quartz Composer) promueven los datos como concepto principal detrás de cualquier programa. La ejecución de un datagrama (Figura 11) es dirigida por el dato o dependiente del mismo. El flujo de datos entre los nodos del programa, líneas no secuenciales de texto, determina el orden de ejecución.

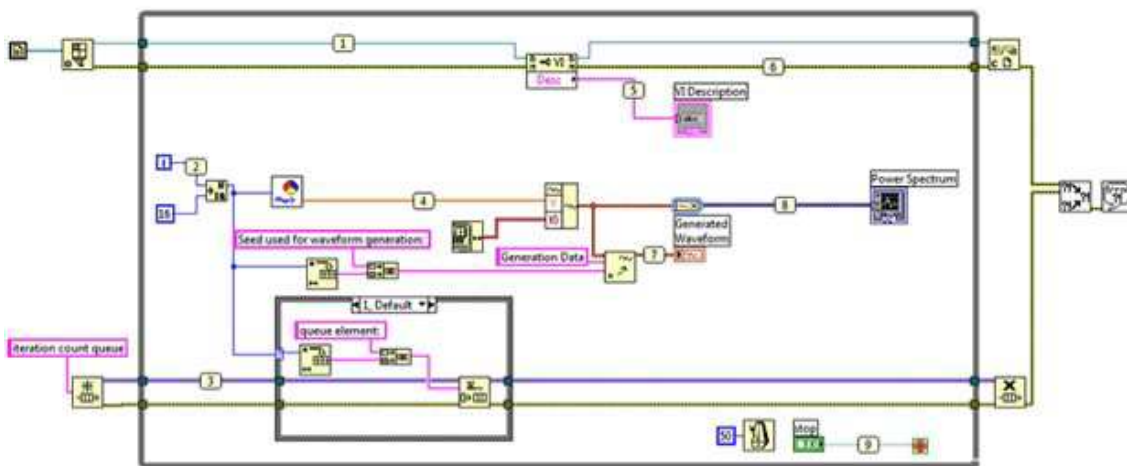


Figura 11: Programación en LabVIEW

La facilidad de uso de estos lenguajes gráficos es precisamente la abstracción de las tareas de bajo nivel. LabVIEW no se queda únicamente en ofrecer un entorno visualmente atractivo, o más intuitivo a la hora de programar, sino que la programación G automáticamente trata muchos de los desafíos a los que debe enfrentarse el usuario en un lenguaje basado en texto como el uso de memoria. En un lenguaje de texto, el usuario es responsable de reservar memoria antes de usarla y liberarla cuando no se precise. Una reserva inadecuada de la memoria es un problema complejo de depurar, pero gracias a este tipo de programación el usuario no debe preocuparse de ello.

Además de todas estas ventajas, el entorno de programación LabVIEW, tiene la capacidad de integrarse con multitud de lenguajes de programación ya existentes. Se pueden crear partes de código escrito (de modo similar a C), es capaz de interpretar código ya desarrollado en MathScript (archivos .m típicamente desarrollados para Matlab), acceso a bibliotecas, código .NET, y multitud de opciones más.

2.5 ROBOTS DE INVESTIGACIÓN Y SENSORES

Por último en este capítulo queda hacer una breve mención a la robótica y los sensores, ya que se hará uso de ellos en el transcurso del proyecto.

Enobio no es el único dispositivo en el mercado que se comercializa para la lectura de biopotenciales fuera de los laboratorios. Al menos otros dos dispositivos, el Epoc de la empresa Emotiv[21] y el Mind Wave[22] de la empresa Neuro Sky. El Epoc, es un dispositivo que dispone de 14 canales en vez de los 4 de los que dispone Enobio, su comunicación es inalámbrica y su precio ronda los cientos de euros. Es un dispositivo cuyo objetivo es proporcionar a usuarios y desarrolladores la posibilidad de trabajar con este tipo de sensores, investigar y encontrar nuevos posibles usos. El Mind Wave (Figura 12) por otra parte es un dispositivo más sencillo que Enobio y Epoc. Dispone únicamente de un canal, su transferencia de datos es también inalámbrica y funciona con pilas, con una autonomía de 10 horas. Sin embargo este Mind Wave no está pensado para desarrolladores, es un hardware para usar con unos ciertos programas que comercializa la empresa que lo distribuye de forma que se utiliza como un “Tutor de educación”. El precio de este Mind Wave es de 99 dólares.



Figura 12: Dispositivo Mind Wave de NeuroSky[22]

Se han desarrollado muchas plataformas de aprendizaje para el estudio de la robótica y las comunicaciones. Son típicos en ambientes de aprendizaje, quizá ya a un nivel universitario, los sistemas robóticos gobernados por placas que poseen la habilidad de ser reprogramadas, como por ejemplo Arduino gobernada por un microcontrolador, o pequeños ordenadores (*roboboards*) sobre los que funciona un sistema operativo en tiempo real. A lo largo de este proyecto se utilizará un robot llamado DaNI, comercializado por Pitsco [23], y en cuyo desarrollo ha colaborado National Instruments. Además todo el control del sistema robótico se lleva a cabo a través de una placa sbRIO 9631 de National Instruments.

Por otra parte, los sensores se han convertido en algo con lo que interactuamos cada día casi sin darnos cuenta. Los teléfonos móviles por ejemplo, y más los de última generación (*smart phones*), utilizan sensores para conocer su estado y otorgarnos la información que más nos puede convenir en cada momento. Sensores de luminosidad, de presión, táctiles, acelerómetros, GPS, etc. Y cada vez desde más temprana edad los niños interactúan y se acostumbran a utilizarlos y a aprender cómo

reaccionan estos sensores. Un ejemplo muy claro de esto es la última consola de Nintendo, la Wii (Figura 13). Esta consola es manejada mediante un mando que a parte de los habituales botones, posee una cámara de infrarrojos capaz de medir en el entorno de la pantalla de televisión a qué punto se dirige el mando; también posee varios giróscopos y acelerómetros, con los que cualquier niño de 10 años seguramente gane en un partido de tenis virtual a un adulto. También se hará uso de este mando a lo largo del proyecto, ya que este tipo de hardware se ha introducido tanto en nuestra cultura que la mayoría de personas podrían utilizar un sistema basado en este control sin necesidad de un manual de instrucciones. Esto es, que el usuario no se tiene que acoplar a la máquina o robot que pretende utilizar, sino que la interacción se realiza de forma natural.



Figura 13: Mando Wii

3. HERRAMIENTAS Y PLATAFORMA UTILIZADA

Antes de comenzar la explicación detallada de cada una de las partes que conforman este proyecto, a continuación se pueden ver unos esquemas que ayudarán al lector a entender mejor lo que se explica a continuación.

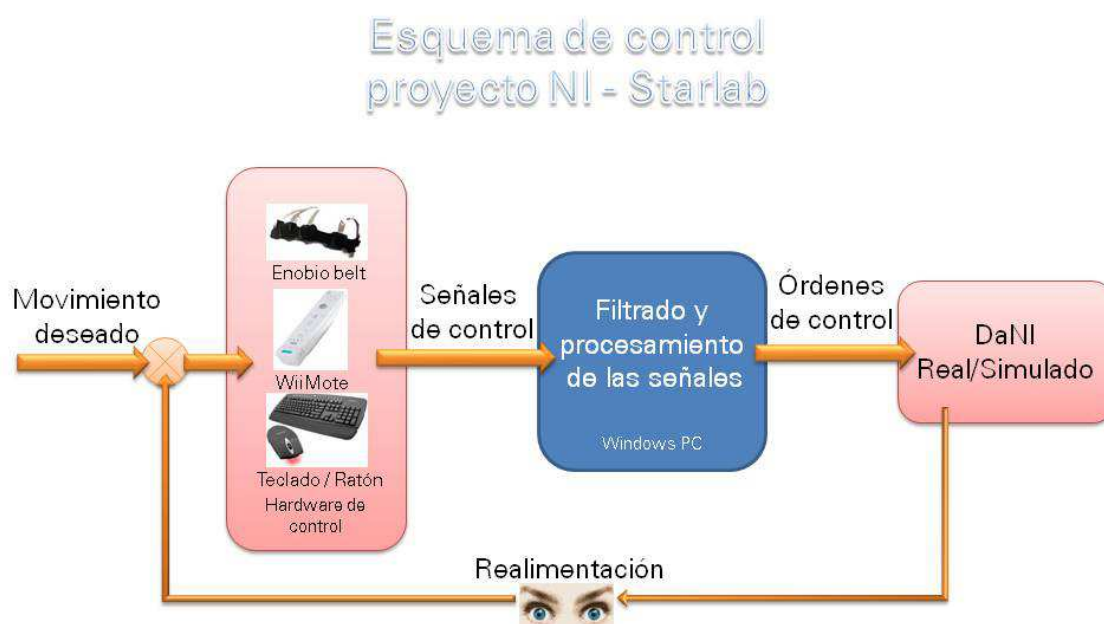


Figura 14: Esquema de control del proyecto

En el esquema de la Figura 14 se puede observar el método de control del robot, y a continuación en la Figura 15 se puede ver un esquema en el que se distinguen los diferentes enlaces entre el software y el hardware utilizado.

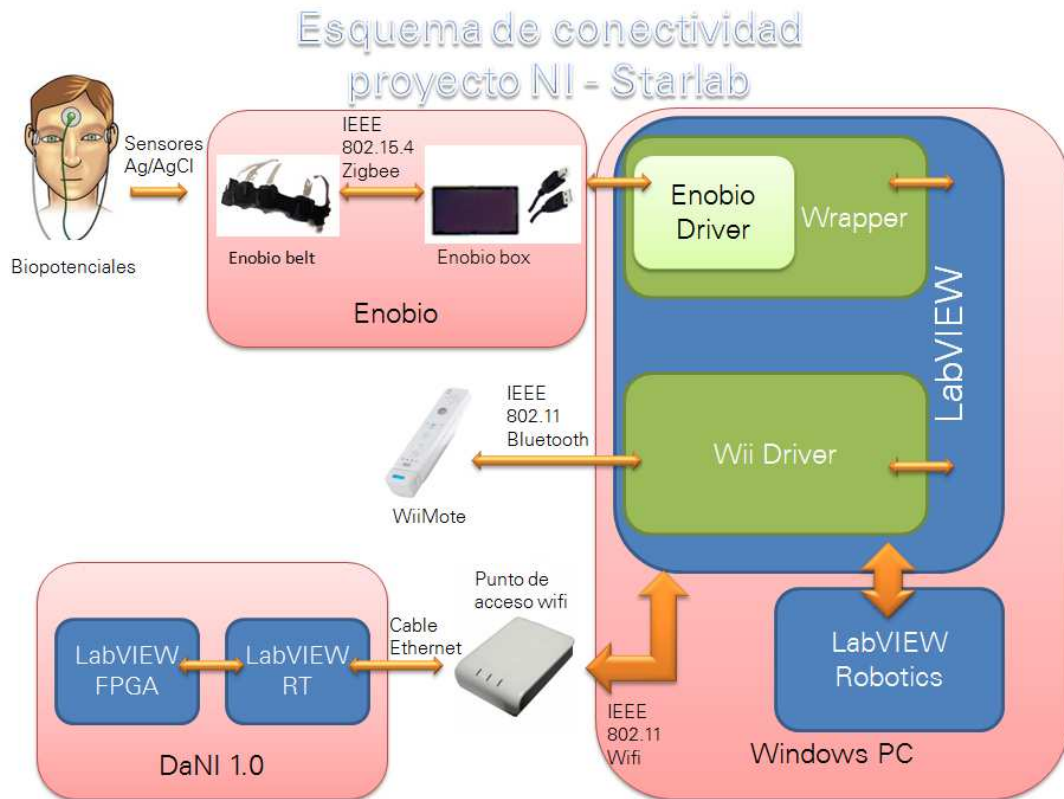


Figura 15: Esquema de conectividad del proyecto

3.1 ENOBIO

Enobio ® es un sistema de sensores electro-fisiológicos modular e inalámbrico preparado para la adquisición de señales EEG, ECG y EOG:

- EEG, electroencefalograma (actividad cerebral).
- ECG, electrocardiograma (actividad del corazón).
- EOG, electrooculograma (movimiento ocular).

Las especificaciones técnicas del hardware se detallan en la Tabla 2:

Número de canales	4	Duración de baterías	8 horas
Salida de datos	Codificación ASCII o EDF y/o <i>streaming</i> digital sobre TCP/IP	Entradas	Biopotenciales (EEG, ECG, EOG)
Ancho de banda	125 Hz	Tiempo de muestreo	250 S/s
Comunicación inalámbrica	IEEE 802.15.4	Rango dinámico efectivo	22 bits
Dimensiones	66 x 55 x 25 mm	DC Offset	Compensación automática del offset para cada canal
Peso	65 gr.	Rechazo a modo común	96 dB

Tabla 2: Especificaciones técnicas de Enobio

Una de las ventajas de Enobio es que sus sensores pueden ser utilizados en seco (sin una solución conductora entre la piel y el sensor). Aunque sí que mejora el CMRR al aplicarla sobre el electrodo DRL. Y mejora la inmunidad a los artefactos.

Además el sistema no necesita ninguna preparación específica de la piel para el uso. La única preparación necesaria es colocar los sensores de forma que haya un buen contacto entre los electrodos y la piel, y una pequeña cantidad de gel para el electrodo DRL.

La señal más pequeña que puede medir Enobio está determinada por el ruido de amplificación, y para bajas frecuencias esta señal es de 0.5 μV p-p. Y la resolución de Enobio es de 0.589 μV .

El ruido introducido en las señales, debida a la electrónica del propio aparato es de 10 a 100 μV .

Enobio está dotado de un amplificador analógico, y para poder comparar el rango dinámico en bits del sistema con otro que no posea este amplificador, se puede considerar que la resolución efectiva en bits es:

$$\text{round}(\log_2(70 * 2^{16})) = \text{round}(\log_2(70)) + 16 = 6 + 16 = 22$$

Y la impedancia de entrada de los electrodos es de 62 GOhms, aunque el CMRR es una mejor medida de la calidad de los sensores.

Este hardware se comunica con la Enobio Box mediante el protocolo estándar IEEE 802.15.4, transmitiendo los valores medidos de cada uno de los 4 canales cada 4 ms. Junto con una medida del tiempo transcurrido en milisegundos desde las 00:00:00 del 01 Enero 1970 UTC.

TELEOPERACIÓN DE UN ROBOT TERRESTRE MEDIANTE BIOPOTENCIALES

Posee las siguientes certificaciones para su uso:

- UNE-EN 61326-1:2006 [24]
- UNE-EN 55011:2008+A2:08 (emisiones de radiación) [25]
- UNE-EN 61000-4-2:97+A1:99+A2:01+E:04 [26]
- UNE-EN 61000-4-3:07 [27]

La Enobio Box se conecta al ordenador utilizando un puerto USB, y se accede a los datos a través del software proporcionado por Starlab (Figura 16), que permite la visualización de las señales y el volcado a disco en un archivo de texto, o bien haciendo uso de la API desarrollada por ellos mismos en C++ o en Java.

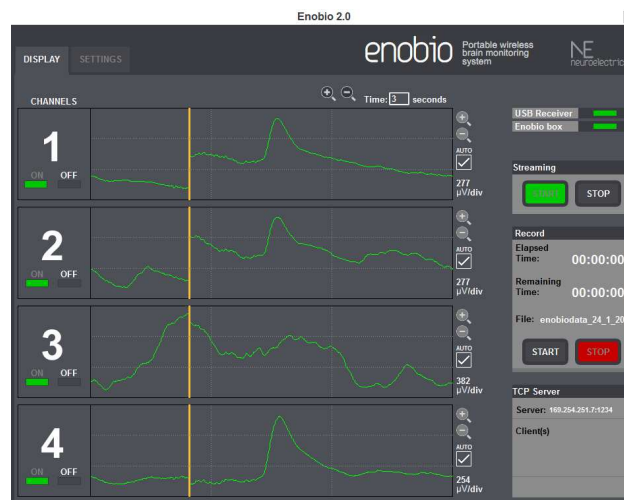


Figura 16: Software Enobio

3.2 DaNI 1.0

DaNI es un robot comercializado por Pitsco, en colaboración con National Instruments. Consta de chasis, ruedas, tren de potencia, motores, sensores, control mediante computadora y cableado. El hardware puede ser estudiado mediante ingeniería inversa y modificado por el usuario. Está pensado para la educación centrándose sobre todo en fundamentos de percepción y control, implementándolos con el software de desarrollo LabVIEW.

En la Figura 17 se puede observar un diagrama del montaje de DaNI.

LabVIEW Robotics Starter Kit (Block Diagram)

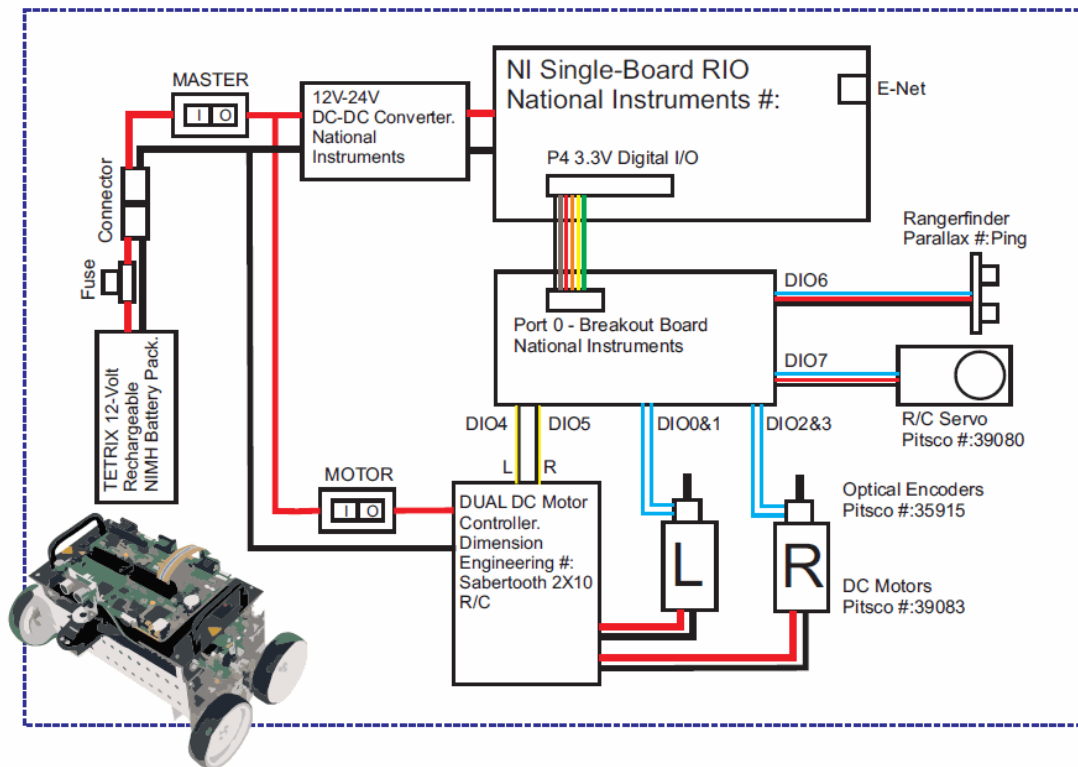


Figura 17: Diagrama de montaje de DaNI[28]

Las partes principales del robot son, como centro de computación una placa sbRIO 9631, una batería NI-MH de 12 V que otorgará autonomía al robot, un controlador para los dos motores DC que moverán las ruedas, encoders que darán información sobre la posición de cada motor, un servo y un sensor de ultrasonidos.

El sensor de ultrasonidos es un Parallax PING, y cuyas principales características se detallan en la Tabla 3.

Tensión de alimentación	5 VDC
Corriente de alimentación	30 mA tip; 35 mA máx.
Rango	2 cm a 3 m
Trigger de entrada	Pulso positivo TTL, 2 μ S min, 5 μ S tip
Pulso de eco	Pulso positivo TTL, 115 μ S a 18.5 ms
Espera de eco	750 μ S desde la caída del pulso de <i>trigger</i>
Frecuencia de ráfaga	40 KHz durante 200 μ S
Indicador de ráfaga	LED indica la actividad del sensor
Retardo entre medidas	200 μ S
Medidas	22mm x 46mm x 16 mm

Tabla 3: Características del sensor ultrasónico

Los motores de corriente continua se alimentan a 12 V, y tienen una velocidad máxima de 152 RPM.

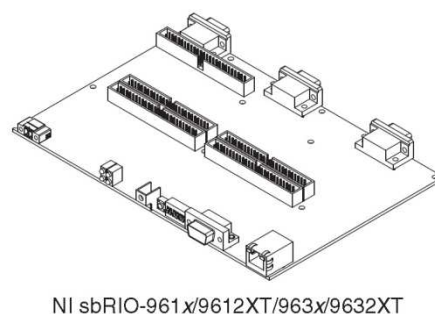
Los encoders se alimentan a 5 V, son encoders de cuadratura de 400 pulsos por revolución.

3.2.1 sbRIO 9631

Esta placa, fabricada por National Instruments, es un dispositivo embebido de control y adquisición, integra un procesador en tiempo real, una FPGA reconfigurable por el usuario y E/S en una sola tarjeta de circuito impreso (PCB). Tiene un procesador industrial de 266 MHz, un FPGA Xilinx Spartan de 1M de Compuertas, 110 líneas de E/S digital de 3.3 V (tolerancia de 5 V / compatible con TTL), 32 canales de entrada analógica de un solo terminal y 16 diferenciales de 16 bits a 250 KS/s y cuatro canales de salida analógica de 16 bits a 100 KS/s. También tiene tres conectores para E/S de expansión usando módulos de E/S de la Serie C a nivel de tarjeta. La sbRIO-9631 ofrece un rango de entrada de suministro de potencia de 19 a 30 VDC. Brinda 64 MB de DRAM para operación embebida y 128 MB de memoria no volátil para almacenamiento de programas y registro de datos. Y funciona sobre un sistema operativo en tiempo real VxWorks.

Este dispositivo tiene un puerto de Ethernet de 10/100 Mbits/s que se puede usar para llevar a cabo comunicación programática en la red y Web integrada (HTTP) y servidores de archivos (FTP). Y posee un puerto serie RS232 para controlar dispositivos periféricos.

En la Figura 18 se puede ver un dibujo de la placa sbRIO 9631.



NI sbRIO-961x/9612XT/963x/9632XT

Figura 18: Dibujo de sbRIO 9631[28]

Para más información acerca de este hardware, ver la documentación incluida en el anexo.

3.3 LabVIEW

LabVIEW es un entorno de desarrollo que brinda a científicos e ingenieros integración con hardware sin precedentes y amplia compatibilidad. LabVIEW inspira a resolver problemas, acelera la productividad y da la seguridad para innovar continuamente para crear y desplegar sistemas de medidas y control.

3.3.1 INTRODUCCIÓN AL ENTORNO DE DESARROLLO

Los programas de LabVIEW son llamados instrumentos virtuales o VIs ya que su apariencia y operación imitan a los instrumentos físicos, como osciloscopios y multímetros. LabVIEW contiene una extensa variedad de herramientas para adquirir, analizar, visualizar y almacenar datos, así como herramientas para ayudar al programador a solucionar y depurar problemas en el código que escriba.

Cuando se crea un nuevo VI, este se divide en dos ventanas, el panel frontal del VI y el diagrama de bloques del VI.

El panel frontal es la interfaz de usuario para el VI, y es donde se colocan los controles e indicadores que el usuario podrá ver y con los que podrá interactuar durante la ejecución del mismo. Podrá acceder a estos durante la programación a través de la paleta de controles.

El diagrama de bloques contiene además de los terminales de control e indicadores, subVIs, funciones, constantes, estructuras y cables.

Los subVIs son Vis que se crean para utilizarlos dentro de otro VI, se utilizan para encapsular código y para hacer reutilizable parte de código que se repite en ese o diferentes proyectos. Ayudan también a crear diagramas de bloques más limpios y mejor estructurados, VIs escalables y facilita el mantenimiento los proyectos. Trabajan de manera similar a funciones en lenguajes de programación textual.

La paleta de funciones está dividida en categorías, una primera categoría con nodos funciones y estructuras de programación, y posteriormente una serie de categorías que se pueden ampliar con módulos y que forman una extensa librería de subVIs con código ya desarrollado en LabVIEW.

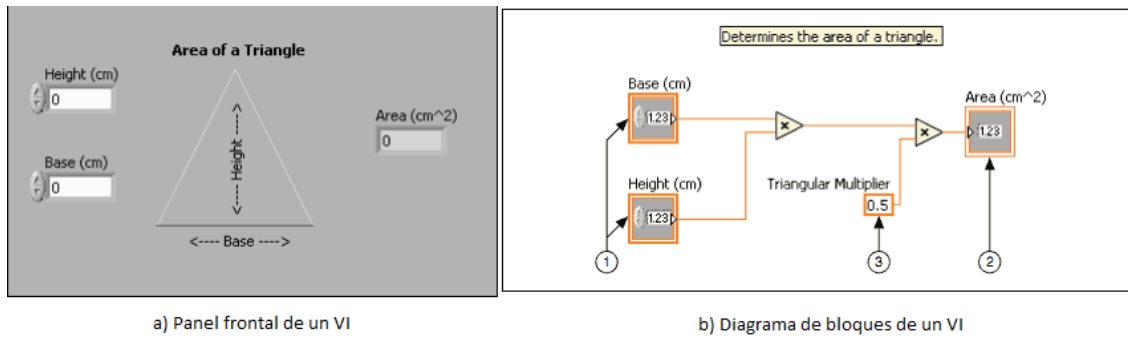


Figura 19: Panel frontal y diagrama de bloques de un VI

En la Figura 19 se puede ver el aspecto del panel frontal y el diagrama de bloques de un VI que calcula el área de un triángulo. En el diagrama de bloques pueden verse referenciados con un 1 los controles, con un 2 un indicador y con un 3 una constante.

3.3.2 LabVIEW ROBOTICS

NI LabVIEW Robotics (Figura 20) proporciona una solución estándar para desarrollo de hardware y software para diseñar sistemas de control de robótica. El Módulo NI LabVIEW Robotics se conecta al entorno de desarrollo de LabVIEW y ofrece una extensa biblioteca con conectividad integrada con sensores de robótica y actuadores, algoritmos básicos para operación inteligente y percepción robusta y funciones de movimiento para mover robots o vehículos. LabVIEW Robotics es ideal para generar prototipos de:

- Vehículos autónomos y semi-autónomos incluyendo sistemas agrícolas y militares
- Plataformas de robots de rescate
- Vehículos submarinos y aéreos
- Robots personales y de servicio
- Dispositivos médicos de robótica
- Robots académicos y de investigación

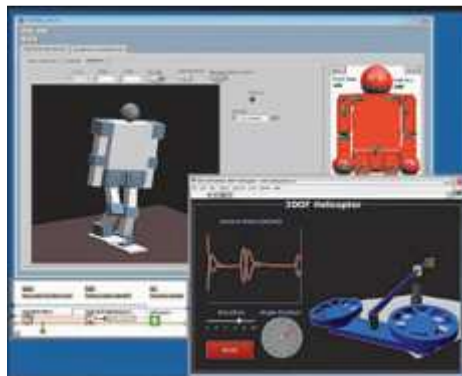


Figura 20: LabVIEW Robotics[28]

Ingenieros, investigadores y educadores en todo el mundo han usado el software de diseño gráfico de sistemas NI LabVIEW para crear aplicaciones de robótica desde robots académicos hasta vehículos autónomos y robots industriales de base fija. La habilidad para importar desde otros lenguajes - incluyendo C y HDL - y para comunicar con una amplia variedad de sensores usando controladores integrados reduce drásticamente el tiempo de desarrollo, así el usuario puede enfocarse en añadir sus propios algoritmos e inteligencia. Esta plataforma es capaz de ejecutarse en una variedad de procesadores incluyendo sistemas comerciales en tiempo real, y basados en FPGA, así como microprocesadores y microcontroladores personalizados. Este paquete incluye los módulos LabVIEW Real-Time, LabVIEW FPGA, Vision Development, LabVIEW Control Design and Simulation y LabVIEW NI SoftMotion.

3.3.3 LabVIEW RT

El módulo NI LabVIEW Real-Time es una solución completa para crear sistemas embebidos, autónomos y confiables con un enfoque de programación gráfica. Como un complemento al entorno de desarrollo de LabVIEW, el módulo LabVIEW Real-Time ayuda a desarrollar y depurar aplicaciones gráficas que pueden descargarse y ejecutarse en dispositivos de hardware embebidos como NI Compact RIO, NI Single-Board RIO, PXI, sistemas de visión o hasta PCs de terceros.

Aunque LabVIEW se usa comúnmente para desarrollar aplicaciones que se ejecutan en sistemas operativos de PC como Windows, estos sistemas operativos no están optimizados para ejecutar aplicaciones críticas durante un periodo de tiempo extendido. Se puede usar el mismo enfoque gráfico de programación en LabVIEW con el módulo LabVIEW Real-Time para crear sistemas autónomos que se ejecutan en periodos de tiempo extendidos.

Con el módulo LabVIEW Real-Time, se pueden aprovechar cientos de bibliotecas de LabVIEW pre-escritas; incluyendo control de algoritmos PID (proporcional integral y derivativo) y FFTs (Transformadas Rápidas de Fourier) en los sistemas autónomos.

3.3.4 LabVIEW FPGA

El módulo NI LabVIEW FPGA (Figura 21) extiende la plataforma de desarrollo gráfico de LabVIEW para descargar código a FPGAs (*Field Programmable Gateway Arrays*) en hardware de E/S reconfigurable (RIO). LabVIEW es ideal para programación FPGA ya que representa paralelismo en flujo de datos, así los usuarios tanto experimentados como principiantes en el diseño tradicional FPGA pueden aplicar de manera productiva la potencia del hardware reconfigurable.

TELEOPERACIÓN DE UN ROBOT TERRESTRE MEDIANTE BIOPOTENCIALES

Tradicionalmente las FPGAs han sido programadas usando VHDL o Verilog. Muchos ingenieros y científicos no están familiarizados con estos lenguajes complejos o requieren una herramienta que les de mayor productividad en diseño a un nivel más alto de abstracción para simplificar significativamente el proceso de generar código FPGA. LabVIEW FPGA es más intuitivo y fácil a la hora de comenzar a trabajar que las herramientas alternativas.

Las FPGAs son esencialmente circuitos de hardware; no hay SO o hilos. Los diseños de LabVIEW que requieren procesamiento de alta velocidad, paralelismo, fiabilidad y estrecho determinismo son perfectos para descargar a FPGAs.

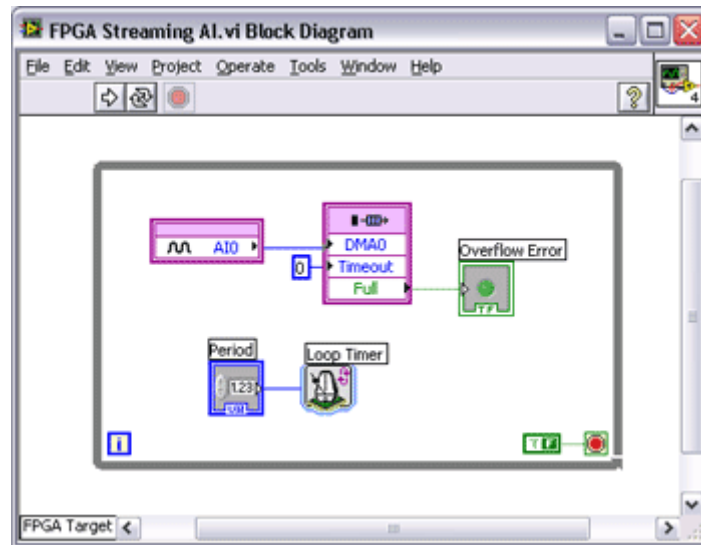


Figura 21: LabVIEW FPGA

3.4 DIAdem

Para el estudio de los datos adquiridos se ha hecho uso de DIAdem (Figura 22), una sola herramienta de software que se puede usar para ubicar, cargar, visualizar, analizar y crear reportes rápidamente de los datos de medidas obtenidos durante la adquisición o generados durante simulaciones. Está diseñado para cumplir con las demandas de hoy en día, puede tener acceso y procesar grandes volúmenes de datos en múltiples formatos para tomar decisiones respaldadas con información.

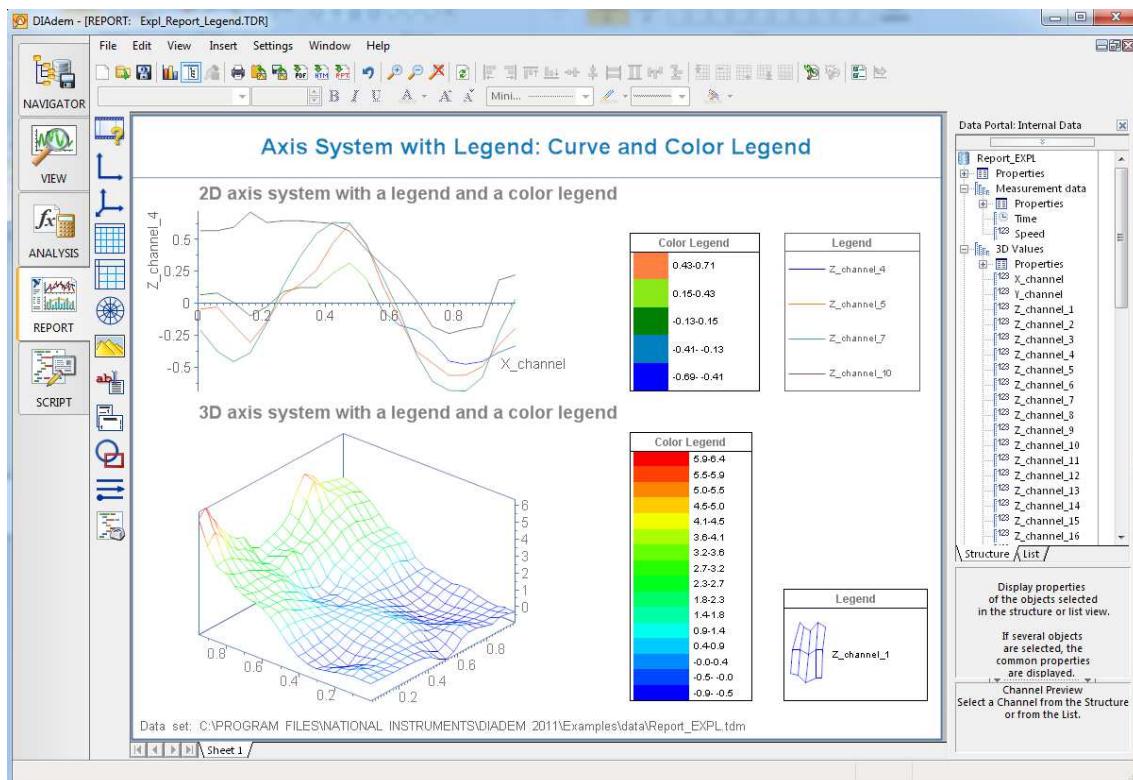


Figura 22: Ventana de reporte de DIAdem

Los puntos clave de este software son:

- Administración de datos y extracción:
Se pueden crear búsquedas avanzadas para encontrar rápidamente datos ubicados en un disco o la red.
- Pantallas de visualización interactivas:
Se puede ver la reproducción de los datos con videos, modelos en 3D, mapas GPS, etc. Se pueden cargar y combinar canales de medida de múltiples archivos para una comparación inmediata.
- Cálculos avanzados de datos:

TELEOPERACIÓN DE UN ROBOT TERRESTRE MEDIANTE BIOPOTENCIALES

Posee una biblioteca de cálculos de análisis que facilita cientos de cálculos de ingeniería, desde matemática básica hasta procesamiento de señales y más, y no requiere programación.

- Reportes personalizados:
Se pueden crear rápidamente plantillas de reportes.
- Tareas automatizadas de análisis y reportes:
Se pueden capturar los pasos que realiza el usuario de forma interactiva conforme realiza un script, o se puede crear un código usando el Visual Basic Script integrado.

4. INTERCOMUNICACION Y PROGRAMACION

4.1 ENOBIO – LabVIEW

En esta primera parte del proyecto se ha tenido que abordar el capítulo más innovador, ya que se ha desarrollado la compatibilidad entre dos piezas de software independientes y que no estaban específicamente desarrolladas para funcionar la una con la otra.

Este enlace entre LabVIEW y el hardware de Enobio se ha realizado a través de la API proporcionada por la empresa Starlab, desarrolladora del hardware. La empresa facilitó las librerías de control del hardware, desarrolladas en C++, y ya compiladas.

LabVIEW es capaz de realizar llamadas a bibliotecas DLL, pero no es compatible con aquellas desarrolladas en C++ ya que no es capaz de entender el lenguaje orientado a objetos que se implementa con este lenguaje. Esto ha obligado al estudio de la API de Enobio para poder desarrollar un *wrapper*, o envoltorio, sobre esta biblioteca.

Un *wrapper* es básicamente en términos de computación una capa software que es capaz de acceder a una capa separada de manera que el usuario pueda interactuar con la última a través de unas ordenes específicas y programadas de manera independiente en el *wrapper* limitando así el acceso, u obligando a acceder de cierta forma específica. Estos *wrappers* se utilizan para múltiples propósitos, desde filtrar el acceso a redes, mantener protocolos de comunicación mientras que se puede modificar una base subyacente, o compatibilizar capas de software, como es el caso de este proyecto.

Se ha creado utilizando el entorno de desarrollo Microsoft Visual C++ 2010 Express (con licencia gratuita), una librería dinámica a la que se ha denominado "LV_Enobio.dll", programada en C, que interactúa con las librerías otorgadas por el fabricante ("EnobioAPI.dll", "EnobioAPI.lib" y "QtCore4.dll").

En la Figura 23 se puede ver un esquema que representa en qué ha consistido la creación de este *wrapper*.

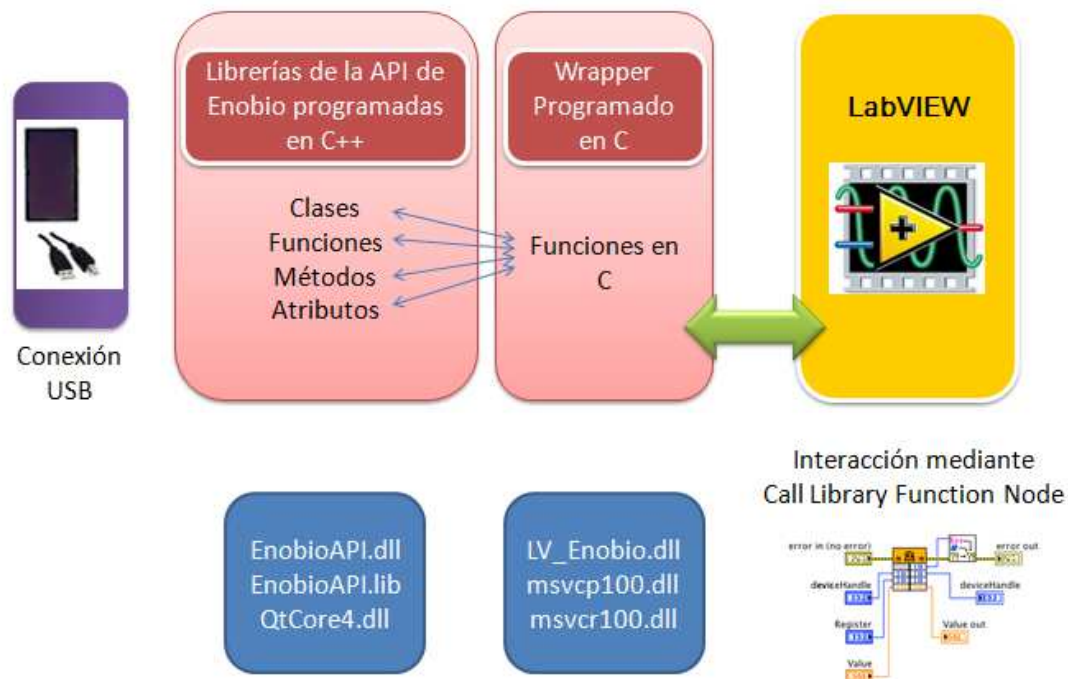


Figura 23: Esquema del enlace LabVIEW - Enobio

Para realizar las llamadas a las funciones publicadas por el *wrapper* se utiliza un "Call Library Function Node". El "Call Library Function Node" es un subVI perteneciente a LabVIEW, llama a DLL o a funciones de librerías compartidas directamente. Es expansible, y muestra los tipos de datos configurados para las entradas y salidas. Soporta múltiples tipos de datos así como diferentes formas de llamada a las funciones. El SubVI consta de pares de terminales de entrada y salida, más un terminal superior con el valor *return* de la función, que queda inutilizado en caso de que éste no exista. Cada par de terminales corresponde a un parámetro de la función, el terminal izquierdo corresponde al valor del parámetro antes de ser ejecutada la función, y el derecho al valor del parámetro después de ser ejecutada. Se deben cablear al subVI todas las entradas y salidas para asegurarse de que se reserva la memoria necesaria, aunque el parámetro sólo sea de salida.

```
int transferir_datos (DatosEnobio *datos_input){
    datos_input->Ch1=ch1;
    datos_input->Ch2=ch2;
    datos_input->Ch3=ch3;
    datos_input->Ch4=ch4;
    datos_input->Tmstp=tmstp_long;
    return 1;
}
```

Figura 24: Función de transferencia de datos del wrapper

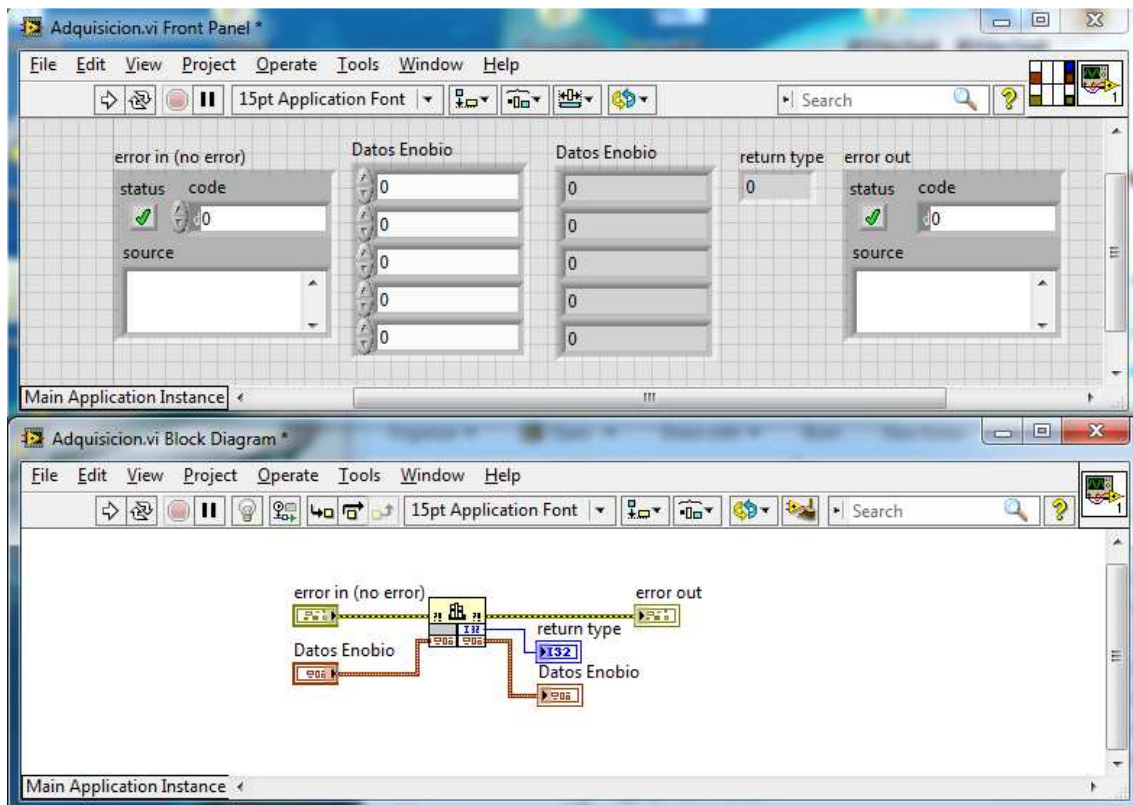


Figura 25: SubVI de adquisición de datos

En las Figura 24 y Figura 25 se pueden observar tanto la función programada que publica el *wrapper*, como el acceso a la misma desde el subVI correspondiente a través de un "Call Library Function Node".

4.2 LabVIEW – LabVIEW Robotics

Parte del proyecto también fue la integración de una simulación 3D del robot DaNI en el proyecto, que posteriormente se utilizaría para comprobar la programación que se iría realizando y validarla sin necesidad de estar haciendo uso del hardware, confiriendo mayor autonomía al desarrollador, permitiéndole trabajar fuera de la oficina donde se encuentra normalmente el hardware. Y además evitando los retrasos que la utilización de una plataforma real puede implicar como son los tiempos de espera de carga de baterías, problemas con conexiones de red, etc. En general cualquier tipo de inconveniente externo al proyecto que pudiera llevar a incrementar el tiempo de desarrollo.

TELEOPERACIÓN DE UN ROBOT TERRESTRE MEDIANTE BIOPOTENCIALES

En este sentido y gracias a las bibliotecas y ayuda que la empresa National Instruments pone a disposición del usuario esta implementación se realizó en unas 10 tardes de trabajo.

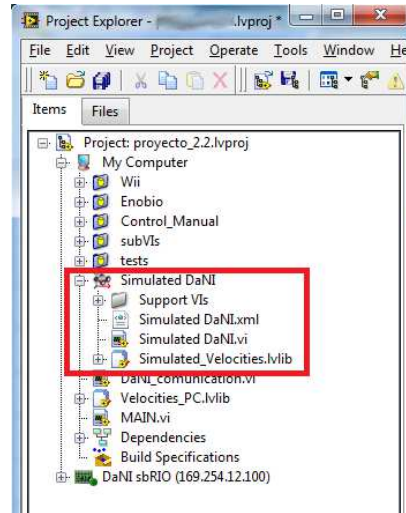


Figura 26: Ventana de proyecto, simulación integrada

En la Figura 26 se puede observar la completa integración de la simulación de DaNI en el proyecto. Sólo se hará uso de ella cuando el usuario así lo requiera, ya que en el mismo proyecto se encuentra la configuración del modelo real del robot.

En la Figura 27 se puede observar una imagen de la simulación en funcionamiento en la fase de desarrollo ya que se aprecia también que el control que se está haciendo del robot es manual como indican los mandos visibles en la parte superior izquierda.

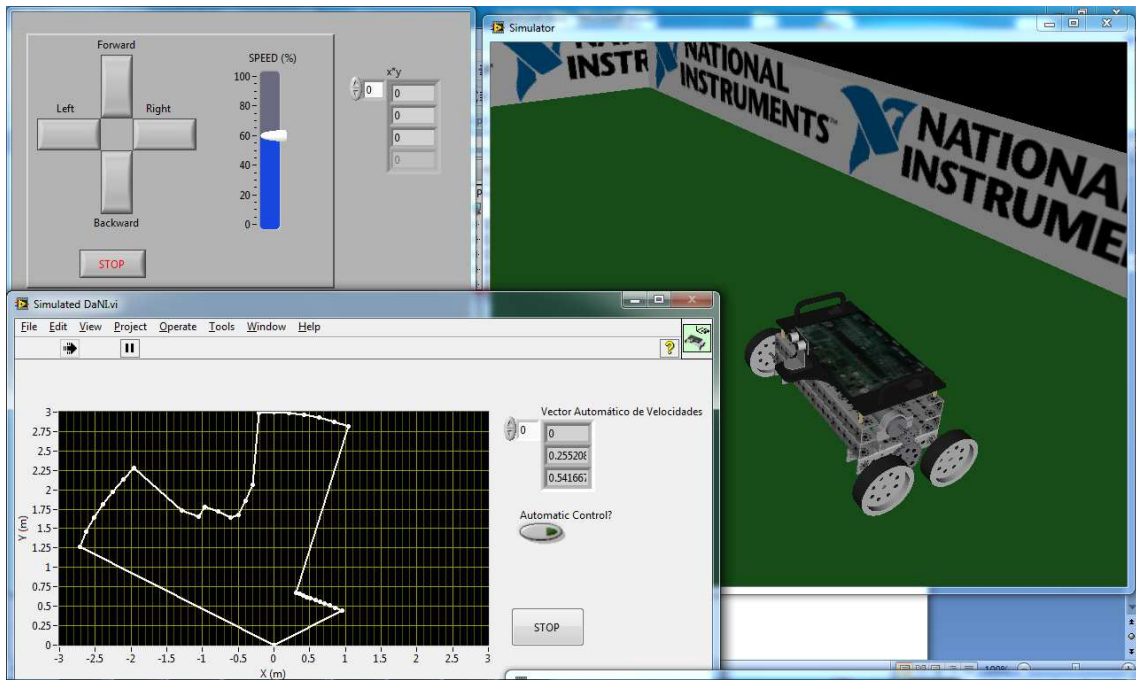


Figura 27: DaNI simulado, control manual

Se puede apreciar también en la parte inferior izquierda un gráfico en dos dimensiones que representa las medidas del sensor de proximidad que el robot lleva instalado en la parte frontal, y es que el modelo de la simulación no sólo es capaz de simular la física del entorno y el robot, sino que es compatible con multitud de sensores cuyas medidas puede obtener también del entorno de simulación gracias a las bibliotecas que incluye el módulo LabVIEW Robotics.

4.3 LabVIEW – Wii Remote

La integración del mando Wii Remote a este proyecto (Figura 28) se propuso como un ejemplo más de compatibilidad de LabVIEW con hardware de terceros, y para mostrar la rapidez de desarrollo que se puede alcanzar con esta plataforma una vez se tiene cierta destreza en el uso de sus posibilidades.

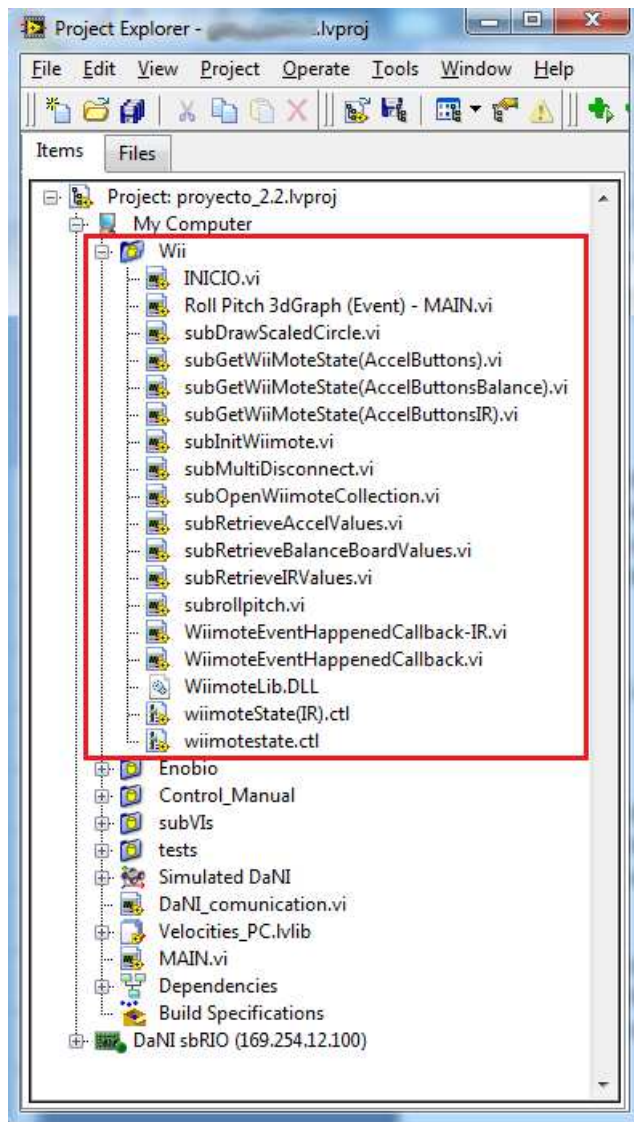


Figura 28: Ventana de proyecto, Wii Remote integrado

Esta parte del proyecto no se tardó en realizar más de 5 días, entre la búsqueda de las bibliotecas abiertas, que se consiguieron de internet [29] y la instalación y uso de las mismas para el control de DaNI. En la Figura 29 se puede ver un esquema de esta conexión.

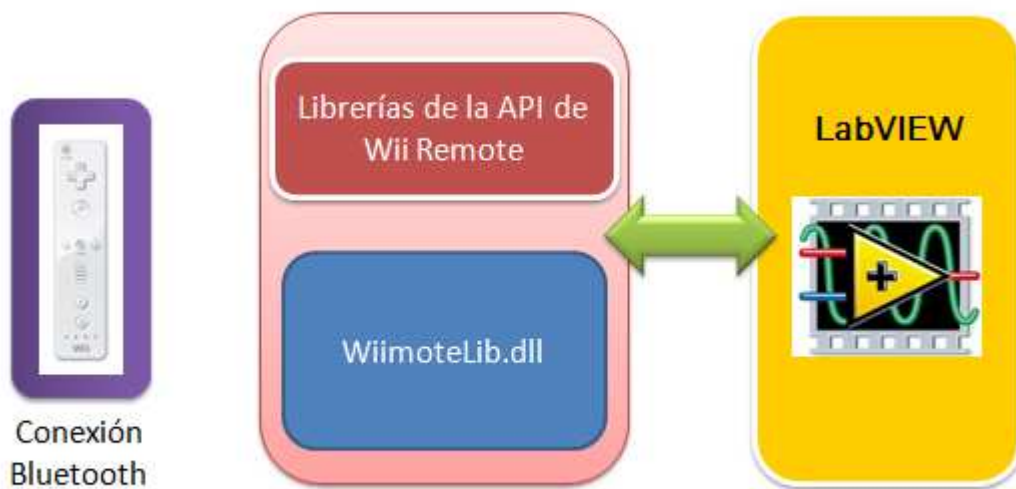


Figura 29: Esquema del enlace LabVIEW - Wii Remote

Para el manejo de la comunicación y eventos se utilizaron unas librerías desarrolladas por la comunidad de usuarios de LabVIEW [30], y la integración final con DaNI y el proyecto a penas fueron unas horas.

4.4 LabVIEW – DaNI

4.4.1 LabVIEW – LabVIEW RT

Como se ha comentado en capítulos anteriores, la plataforma de desarrollo sbRIO 9631, funciona sobre un sistema operativo en tiempo real. Este sistema operativo puede ser programado desde LabVIEW siguiendo los mismos principios que para cualquier otro tipo de programa, lo que facilita su uso al abstraerse el programador de la plataforma que se está utilizando.

Los mismos instrumentos virtuales (VIs, *Virtual Instruments*), así como otras funciones y estructuras usadas en LabVIEW se utilizan también en LabVIEW RT. La diferencia fundamental es que el código generado con este último sistema se puede descargar para ser ejecutado sobre hardware independiente de adquisición de datos en tiempo real. En nuestro caso este hardware es la sbRIO 9631. Y el usuario no ha de preocuparse por descartar o eliminar de los programas cosas como las interfaces de usuario (UI), que vienen a ser los paneles frontales de los VIs, que como es normal ocupan un gran espacio en memoria, y su actualización durante la ejecución ralentizaría los programas. Al ser descargados los programas en el destino RT, LabVIEW elimina todas estas partes de los programas que no van a ejecutarse en el haciendo una instalación “limpia” del código que va a ejecutarse.

La programación RT que se ha realizado en el proyecto ha sido básicamente la modificación de un programa anterior que controlaba DaNI de forma automática de manera que se transmitan continuamente las órdenes de control desde el PC a la placa, y este programa en tiempo real gestione esta información recibida y se la transmita al código de la FPGA que controla el sensor, los motores, encoders, etc. Esta transmisión se realiza mediante un enlace con el programa (VI) compilado en la FPGA, a través de un "FPGA Open VI Reference" (Figura 30), subVI incluido en las librerías del módulo LabVIEW FPGA. Este enlace confiere acceso a los controles e indicadores programados previamente en la FPGA.

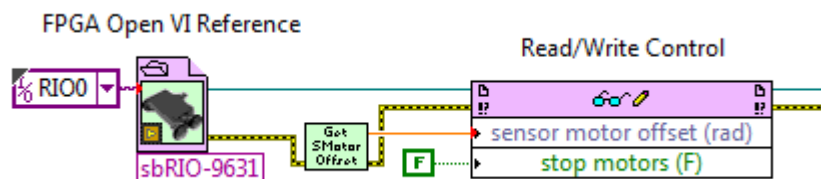


Figura 30: Diagrama de bloques, enlace RT - FPGA

4.4.2 LabVIEW RT – LabVIEW FPGA

La placa de desarrollo sbRIO 9631 como se ha comentado anteriormente posee una FPGA Xilinx Spartan-3, que puede conectarse a sus estradas y salidas o bien puentearlas para que sean accesibles directamente desde el sistema operativo en tiempo real. Gracias al módulo LabVIEW FPGA, se pueden programar las compuertas desde el mismo entorno y utilizando el mismo tipo de programación que para el resto de módulos o aplicaciones, así el programador vuelve a quedar abstraído del hardware que está programando sin necesidad de implementar código en un lenguaje específico para la FPGA.

Toda la programación de la FPGA se hace en un VI igual que cualquier otro programa, eso sí, utilizando los subVIs específicos que se incluyen en el módulo y atendiendo a ciertas reglas, como es por ejemplo, que debido a la naturaleza del hardware que se está programando las variables numéricas no pueden tener una representación en coma flotante sino que cualquier dato que se vaya a enviar a la FPGA debe ser reconvertido a un número de punto fijo (*fixed point*), y todas las variables numéricas en la FPGA deberán tener este formato. Posteriormente se compila en el ordenador de desarrollo desde el mismo entorno de programación, y de ahí se descarga a la FPGA de la placa a través de una conexión Ethernet.

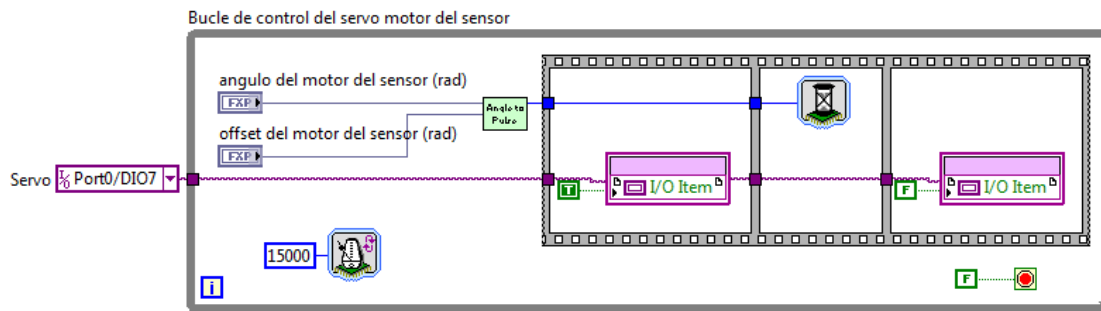


Figura 31: Generación de PWM en la FPGA para el control de un servo

Esta opción permite al usuario programar a nivel de hardware de manera que puede manejar fácilmente salidas como puede ser la generación de PWMs (Figura 31), o la lectura de encoders, creando programas dependientes directamente del reloj de la placa.

5. INTEGRACIÓN Y RESULTADOS EXPERIMENTALES

5.1 ADQUISICIÓN DE DATOS DE ENOBIO

Las bibliotecas suministradas por el fabricante de Enobio están desarrolladas en C++, así que ha sido necesario crear una envoltura sobre estas bibliotecas para hacerlas accesibles desde LabVIEW. Para crear este envoltorio, como ya se ha comentado se ha utilizado el programa Microsoft Visual Studio C++ 2010.

Gracias a la documentación que la empresa facilitó no fue difícil empezar creando pequeñas envolturas de prueba que accediesen al hardware. El estado del mismo, y leer algunas propiedades.

Los primeros problemas se tuvieron a la hora de implementar el patrón productor-consumidor con el que funciona Enobio. Este hardware usa un doble patrón productor-consumidor de datos y estados. Cuando se crea una instancia del hardware en la memoria para poder comunicarse con él, posteriormente hace falta registrar tanto el consumidor de datos como el de estados para que estos puedan ejecutar el código programado en el método correspondiente cada vez que se reciba el evento creado por el productor.

El problema se encuentra en que las bibliotecas están diseñadas de forma que cada vez que el consumidor recibe un evento, se ejecuta un código que llama a un *callback* de manera que éste nuevo dato o estado es recogido por el programa que está llamando a la biblioteca. Pero LabVIEW no puede gestionar este tipo de llamadas, de manera que durante la ejecución del proyecto se comenzó creando un VI (Figura 32) que accediese a la biblioteca y estuviese continuamente leyendo los valores de las últimas variables adquiridas, y posteriormente se filtrarían para eliminar las tomas de datos repetidas.

Este método de obtención de las variables resultó efectivo para poder continuar con el proyecto, pero la velocidad de la lectura de los datos estaba en torno a los 20S/s. Fue suficiente como para obtener la información necesaria para tratarla y llevar a cabo la demostración que se buscaba. En otras palabras, no se consiguió alcanzar los 250S/s. de velocidad de adquisición que brinda el hardware, pero se cumplían las especificaciones requeridas, y trabajando en plazos muy apretados como los que se disponían se decidió seguir avanzando.

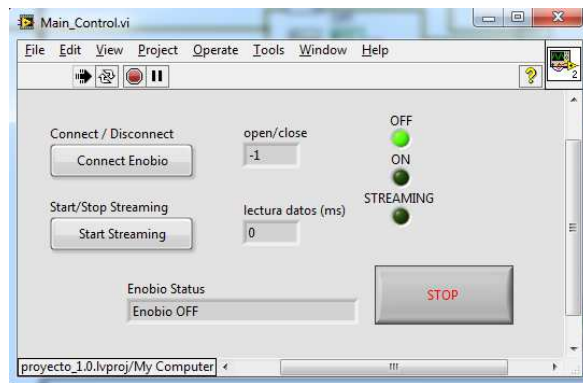


Figura 32: Panel frontal de comunicación con Enobio

Más adelante, avanzado ya el proyecto y cumplidos varios hitos para la finalización del mismo, se propuso intentar mejorar esta adquisición. Se decidió hacer unas modificaciones en la programación de acceso desde LabVIEW, que obligaron a la modificación de parte del código que se compilaba en el *wrapper*. Se creó una estructura en la que se incluían todos los valores a transmitir (Figura 33), y se modificó el método *receiveData*, perteneciente a la clase *EnobioDataConsumer*.

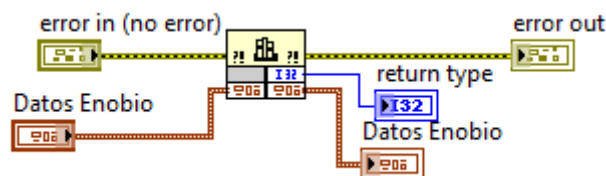


Figura 33: Transmisión de datos entre Enobio y LabVIEW

Junto con este cambio se implementó en LabVIEW lo que realmente permitió que se alcanzaran velocidades de adquisición más altas, y esto fue la creación de una cola a la que accedía, por una parte un VI encargado de la comunicación con las bibliotecas, y por otra parte los VIs que fuesen a hacer uso de los datos introducidos en la cola. El VI encargado de la comunicación con las bibliotecas además realiza una comparación de los valores obtenidos para filtrar los datos repetidos, de esta forma únicamente se introducen en la cola valores tomados por los sensores con su correspondiente etiqueta de tiempo sin que haya datos repetidos. Este cambio no solo ayudó a que la velocidad de adquisición aumentase considerablemente sino que se creó una independencia total entre la adquisición de datos y la utilización de ellos por medio de otros VIs. Esto para la organización del proyecto supuso una ayuda importante. Desde un control principal podíamos ejecutar otros programas que accedieran a los datos con sólo darles acceso a la cola en la que se introducían. De esta forma para la presentación del proyecto en público se pudo crear un programa de test y práctica con su interfaz de usuario, y otro programa de control que ofrecía distinta información útil para cuando el usuario está controlando el robot.

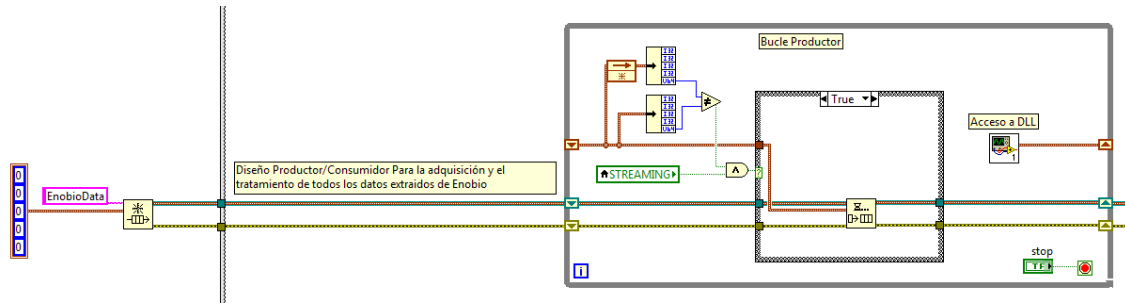


Figura 34: Acceso a biblioteca, adquisición, filtrado y puesta en cola de los datos adquiridos

En la Figura 34 se puede observar cómo se crea la cola y se van introduciendo datos en ella, y en la Figura 35, obtenida de otro VI distinto, se puede ver como se accede a la cola, se van extrayendo los datos de la cola para la obtención de información y la ejecución del control del robot.

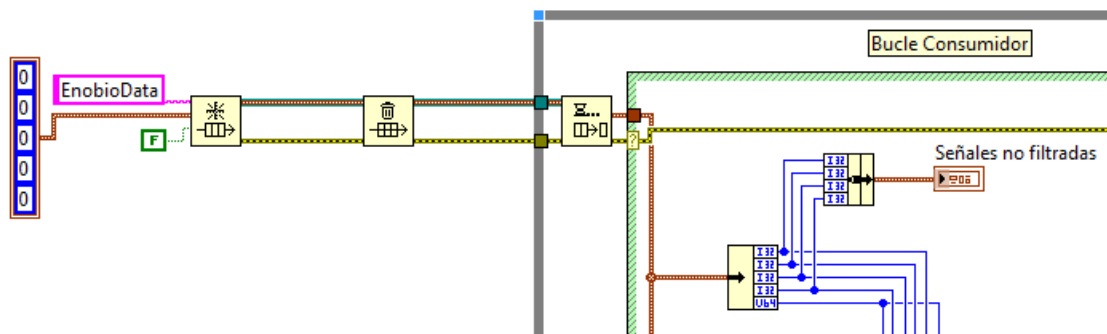


Figura 35: Tratamiento de los datos obtenidos

Estos cambios en el *wrapper* y en LabVIEW permitieron obtener velocidades de adquisición de unos 100S/s por canal.

Actualmente, tras la presentación del proyecto en dos convenciones de la empresa National Instruments, en Madrid y Austria, se ha estado investigando en la posibilidad de añadir una programación de eventos entre LabVIEW y las bibliotecas para intentar adquirir a la máxima velocidad que permite el hardware.

La mejora se ha logrado poco antes de la presentación oficial de este proyecto, mediante la utilización de los eventos de usuario que proporciona LabVIEW al programador. Se ha creado un evento de usuario en el programa que accede a las bibliotecas y adquiere los datos. Este evento se registra dinámicamente haciendo una llamada al *wrapper*, de forma que una vez registrado, cada vez que Enobio Box reciba un dato por vía inalámbrica se registra un evento que se comunica al VI de adquisición, que a su vez hace la llamada a las funciones correspondientes para obtener ese dato. De esta manera la llamada a las bibliotecas se hace únicamente cuando hay un dato nuevo disponible, mejorando el rendimiento del programa puesto que disminuye el uso del procesador, y evitando la necesidad de comprobar si cada dato que se recibe es distinto al anterior.

TELEOPERACIÓN DE UN ROBOT TERRESTRE MEDIANTE BIOPOTENCIALES

A continuación (Figura 36) se puede observar parte del diagrama de bloques donde se crea el evento y se registra en la DLL, así como el acceso a la biblioteca cuando un evento es capturado.

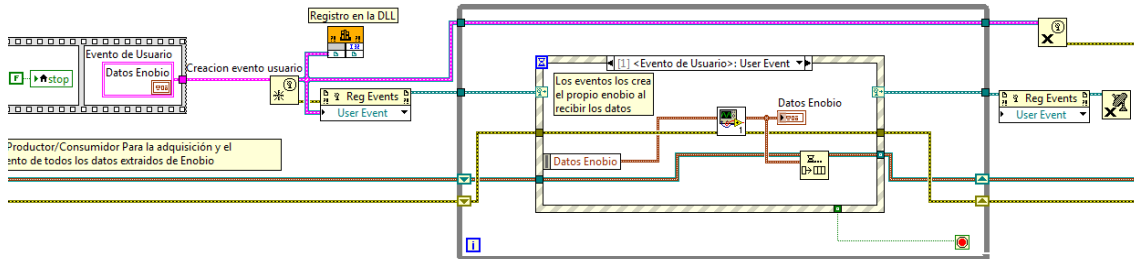


Figura 36: Creación y registro de eventos en el VI principal

Mientras que en la Figura 37 se puede observar parte del código textual que declara y registra los eventos en el *wrapper*.

```
#include <extcode.h>

...

LVUserEventRef *rwer;

int registrar_LVeventos(LVUserEventRef * LVrwer){
    rwer=LVrwer;
    return 1;
}

...

void EnobioDataConsumer::receiveData(const PData &data)
{
    EnobioData * pData = (EnobioData *)data.getData();

    ch1 = pData->getChannel1();
    ch2 = pData->getChannel2();
    ch3 = pData->getChannel3();
    ch4 = pData->getChannel4();
    tmstp_long = pData->getTimestamp();

    PostLVUserEvent(*rwer,(void *)pData
}


```

Figura 37: Funciones de registro y envío de eventos en el *wrapper*

5.1.1 ANÁLISIS DE DATOS CON DIAdem

En el proceso del proyecto, tras la configuración de la conexión y acceso a los datos de Enobio, se procedió a la adquisición de varias muestras de datos y el estudio de los mismos para establecer una estrategia de control según el tipo y calidad de las señales muestreadas. Para esto se realizaron varias tomas de datos, diferentes días, de distinta duración y bajo distintas condiciones modificando los artefactos típicos que podían afectar a las señales.

Casi desde el primer instante se desechó la posibilidad de utilizar las señales EEG y ECG adquiridas por el hardware, debido a la mayor complejidad de su tratamiento y la poca información que se iba a poder obtener de ellas. Sin embargo el EOG medido representaba una buena fuente de información y con una calidad suficiente como para poder filtrarla y realizar el procesamiento necesario para implementar un control del robot.

Sin embargo las señales iban a tener que ser filtradas y realizar algún tipo de conversión en ellas para poder ser utilizadas para el control del robot. En el Gráfico 2 se muestra un gráfico reportado por DIAdem, en el que se ha seleccionado un período de tiempo de unos 200 segundos, en el que se puede observar cómo las señales de los canales están expuestas a cambios que no tienen por qué afectar a los otros canales. Como se ha comentado antes, estos artefactos pueden ser variaciones de la luz incidente en los ojos, movimiento o pérdida de contacto de algún sensor con la piel (como parece que ocurrió transcurrido algo más de 175 segundos en el canal 1). Sin embargo, otros artefactos como los que se observan en los canales 1 y 4 del mismo gráfico que provocan el continuo cambio de la tensión DC en el canal durante cierto tiempo son los que más preocuparon para poder interpretar las señales.

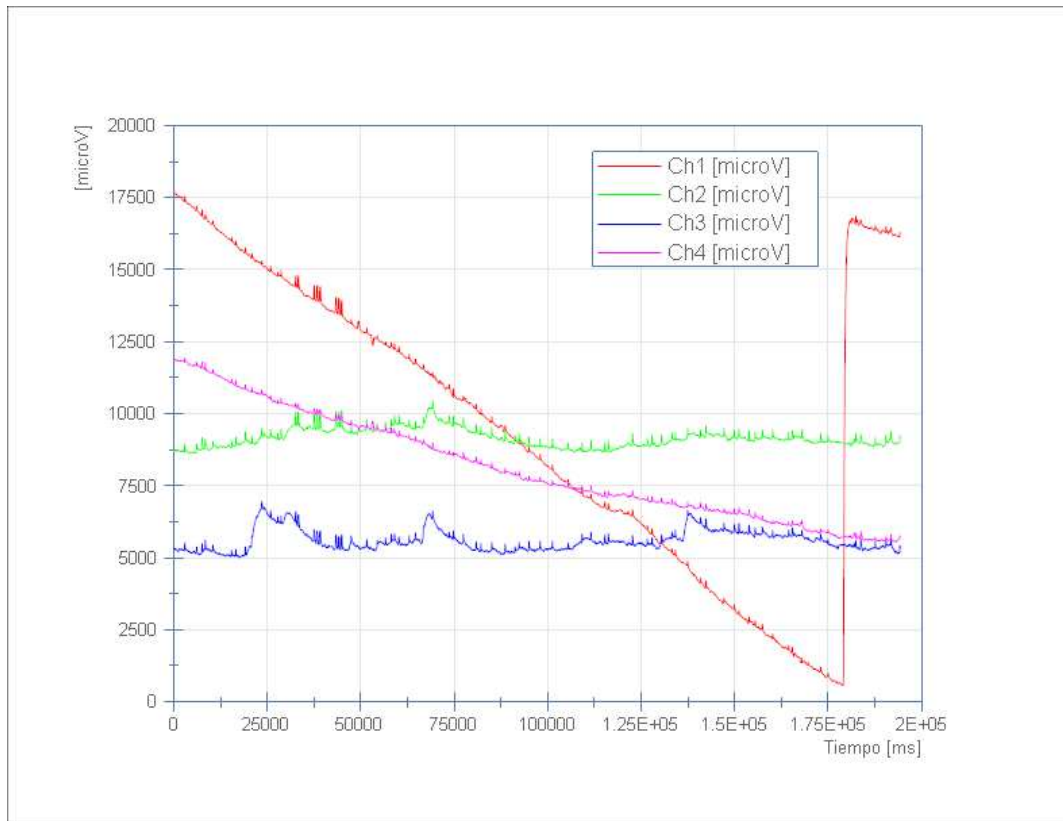


Gráfico 2: Lectura de canales sin filtrar en DIAdem

Se realizaron varios estudios *offline* con las señales muestreadas y utilizando DIAdem. Apoyándose en las herramientas que este software facilita se implementaron distintos tipos de filtros para ver cuál o cuáles de ellos podrían servir para obtener unas medidas sin DC, punto esencial de la estrategia de control que se iba a implementar. Finalmente y dado que en este proyecto la respuesta del procesado de la señal debía ser lo más rápida posible se optó por implementar unos filtros paso alto, aunque con una frecuencia de corte del orden de mHz., eliminando de esta forma la tensión continua de las señales. Esto aunque resuelve el problema de las variaciones de la tensión DC en los canales obliga al no tener una referencia constante a no poder medir con precisión la dirección de los ojos, ni señales constantes como por ejemplo diferenciar entre cuando los ojos están abiertos o cerrados.

5.2 ESTRATEGIA DE CONTROL

Una vez hecho el estudio y decidido cómo iban a tratarse las señales se empezó a diseñar la estrategia de control.

Como primera medida, y para un mejor control del robot se decidió separar los movimientos del mismo en dos grupos independientes. Un primer grupo que consistiría en los movimientos de avance y retroceso del robot, y un segundo grupo que abarcaría los giros a izquierda y derecha. De esta forma se decidió crear dos

máquinas de estados cuyo esquema se puede observar en la Figura 38 que se presentan a continuación.

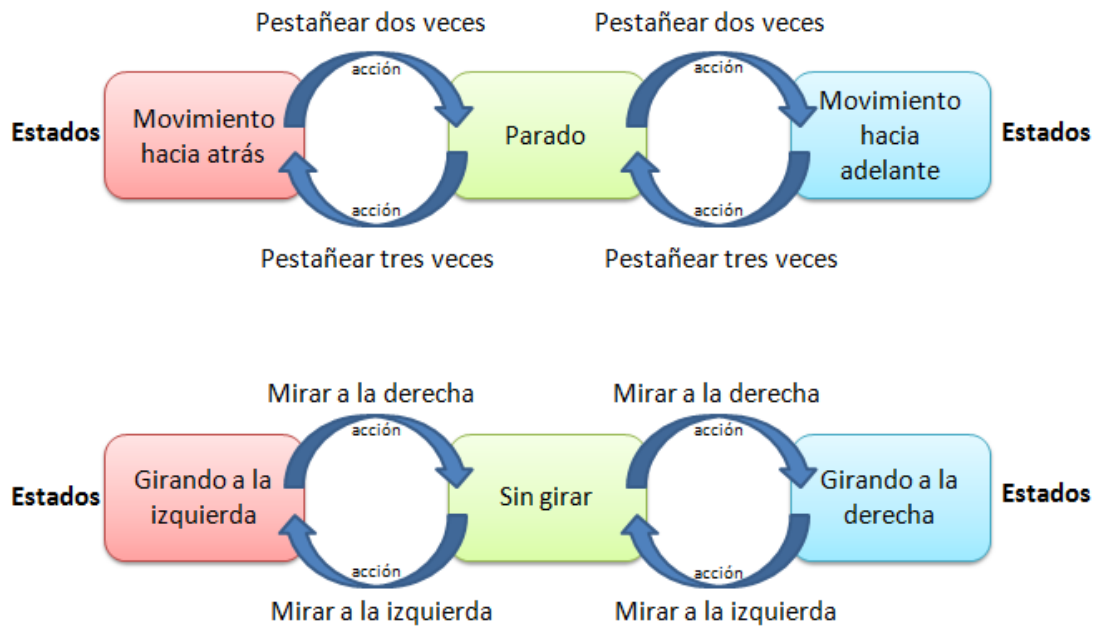


Figura 38: Máquinas de estados

Se tuvo que implementar desde cero todo el software de control para llevar a cabo esta estrategia ya que no existía en ningún sitio, y ha pasado a formar parte importante del “*know-how*” de este proyecto.

Se han utilizado estrategias distintas para las medidas de cada máquina de estados.

Se ha hecho uso de los cuatro canales de que dispone Enobio para la marcha adelante y marcha atrás. Al medirse los pestañeos, que provocan una subida de tensión en todos los canales, la estrategia que se sigue es el sumar los valores de los mismos para poder imponer un límite de detección superior y de esta manera protegerse tanto frente al ruido de los canales como frente a variaciones bruscas que pueda sufrir un canal (como se ha visto que pueden existir en el Gráfico 2 de la página 52). Una vez obtenida esta señal se establecen unos límites basados en el estudio de las medidas que se habían tomado y analizado en DIAdem. Gestionando estas señales se consigue que el usuario sea capaz de comunicar instrucciones al robot prácticamente en tiempo real, sin necesidad de que se deba realizar un entrenamiento extenso previo para el uso del dispositivo. El usuario dispone de aproximadamente un segundo para realizar los pestañeos, por lo que en caso de querer dar una orden, suponiendo que el usuario tardase medio segundo en ejecutar la orden, el robot tardaría medio segundo en responder a dicha orden.

Para los movimientos de giro se utilizan sin embargo dos de los cuatro sensores de que dispone Enobio. Esto es así porque la medida de los sensores es sensible a la localización de los mismos. Los dos sensores localizados en FP1 y FP2 (ver Figura 3 en la página 9) que en la configuración de nuestro proyecto representan los canales 3 y 2 respectivamente, sufren cambios en el valor de su medida cuando se mira hacia los lados, pero debido a su posición durante este movimiento, uno de los ojos induce una señal de tensión más alta, y el otro ojo una de tensión menor en cada sensor, y por lo tanto la medida no es tan representativa como la de los sensores colocados en AF7 y AF8, canales 4 y 1 respectivamente. Estos últimos, al estar afectados cada uno de ellos únicamente por el dipolo que representa el ojo del lado en el que se encuentran, sufren un cambio en su medida mucho más representativo cuando la mirada del usuario se dirige a un lado o al otro. Midiendo así el sensor AF7 (canal4) una tensión más alta y el AF8 (canal 1) una tensión más baja si el usuario mira a la izquierda, y viceversa si mira hacia la derecha. Teniendo en cuenta estas características y siguiendo el mismo planteamiento que para el caso de los pestañeos, para la gestión de estas órdenes lo que se programó es la resta de ambos canales de forma que si el usuario mira a la derecha la tensión resultante es mucho mayor, y si mira a la izquierda mucho menor. En este caso se implementan dos límites de comparación para la señal y así se distingue la orden que se desea enviar. Sin embargo es necesario comentar en este apartado que debido a los filtros implementados y al comportamiento de los sensores, estos movimientos que debe hacer el usuario, necesitan algo de práctica ya que el movimiento que ha de hacerse es un movimiento sacádico para que se detecte la orden, y el programa la compute como una única. El movimiento debe durar entre 0.03 y 0.1 segundos.¹

Esto en cuanto a la estrategia de las máquinas de estados. Sin embargo esta estrategia no sería de ninguna utilidad si las señales no tienen un control o están dentro de un rango. Esto lleva a otra parte importante del programa que es la comprobación y auto calibración del sistema.

Tras el filtrado de las señales, todas ellas por separado y la suma total, se someten a un estudio realizado *online* sobre los últimos 1000 datos adquiridos (esto es aproximadamente los 10 últimos segundos de funcionamiento). En este control se somete tanto a la media como a la varianza de cada uno de los canales y de la suma de ellos a estar dentro de unos límites, determinados con ayuda del estudio de los datos en DIAdem y de la puesta en práctica provocando fallos de medida (como el movimiento de un sensor) para gestionar estos errores. Si 50 medidas consecutivas (0.5 segundos aproximadamente) de un mismo canal se salen del rango establecido entonces este canal se considera no calibrado. Para que el programa considere el correcto funcionamiento de la adquisición de señal todos los canales deben estar calibrados, si cualquiera de ellos no se considera así, fuese cual fuese el estado

¹. Todos estos datos temporales, y los que se refieren a continuación, son referidos al programa de la versión de adquisición denominada 2.3 (lectura constante de datos sin manejo de eventos de usuario). Para la última versión que se ha implementado (mediante eventos), hacen falta algunas pruebas para obtener la mejor calibración, después de lo cual, podrían darse límites e instrucciones de uso mucho más precisos.

anterior del robot, este pasa al estado de reposo donde está parado y sin girar, y tras la calibración de los canales sigue en ese estado salvo que reciba otra orden (no guarda memoria de su estado).

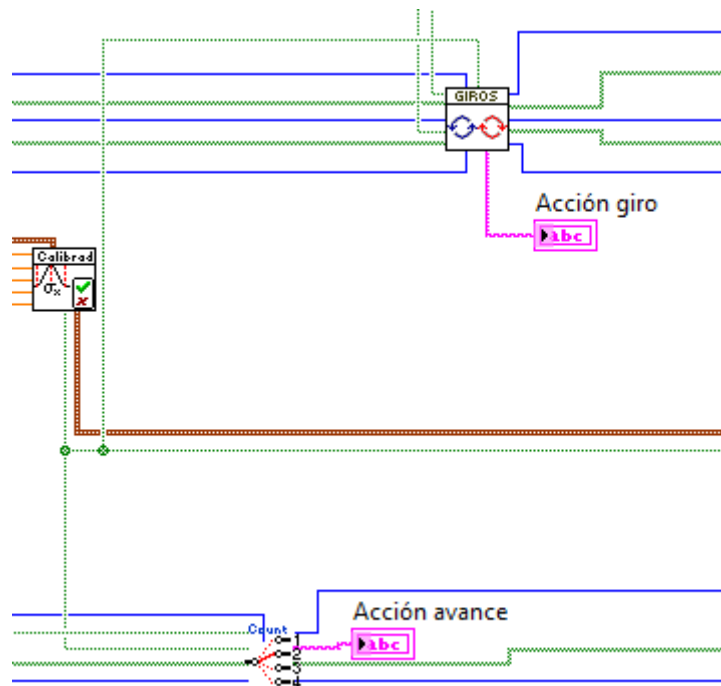


Figura 39: SubVIs de calibración, avance y giro

En la Figura 39 se puede observar como los subVIs que gestionan las dos máquinas de estados poseen una entrada relacionada con el subVI que gestiona el estado de calibración de las señales de Enobio.

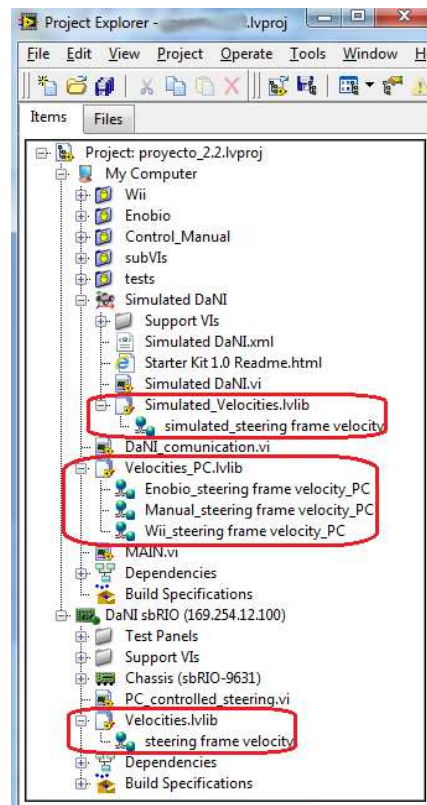


Figura 40: Variables publicadas en el proyecto

Por último, la comunicación. Se desea poder controlar distintos hardwares (o hardware y simulación) desde distintos controles. Se ha desarrollado un sistema de variables compartidas con el mismo formato, cada dispositivo ya sea de control u objetivo, tiene una variable publicada a la que accede el programa principal para leerla o escribirla según se necesite. En la Figura 40 se pueden ver las variables publicadas en el proyecto.

5.3 LAS DIFERENTES OPCIONES DE CONTROL

5.3.1 CONTROL DE LA SIMULACIÓN

Para la simulación en 3D, la integración del entorno es prácticamente automática y National Instruments proporciona el modelo del robot. Se tuvo bastantes problemas y se dedicó bastante tiempo debido a un problema del entorno cuando se trabaja en sistemas operativos configurados con signos de puntuación que no son los estadounidenses. Se conseguía realizar la simulación pero las fuerzas de rozamiento no se tenían en cuenta por lo que el modelo no se movía. Finalmente se accedió a un grupo de desarrollo en este entorno a través de la red interna de la empresa (por lo

que no serviría referenciarlo aquí) y se comentó el problema y proponiendo soluciones se encontró de qué se trataba.

A parte de ese problema, y gracias a que desde un principio se diseñó el proyecto con la intención de trabajar con múltiples controles y objetivos, ya se habían creado unas variables en el proyecto que guardaban las instrucciones que se iban a comunicar al robot objetivo, ya fuese el real o el simulado, por lo tanto se modificó un programa de control del robot simulado para que accediese a estas variables. También se implementó otro sistema de seguridad para el robot consistente en un sistema anti colisión (Figura 41) haciendo uso del sensor de ultrasonidos. Si se detecta un obstáculo muy cercano al robot, este sistema evita que se colisione contra él.

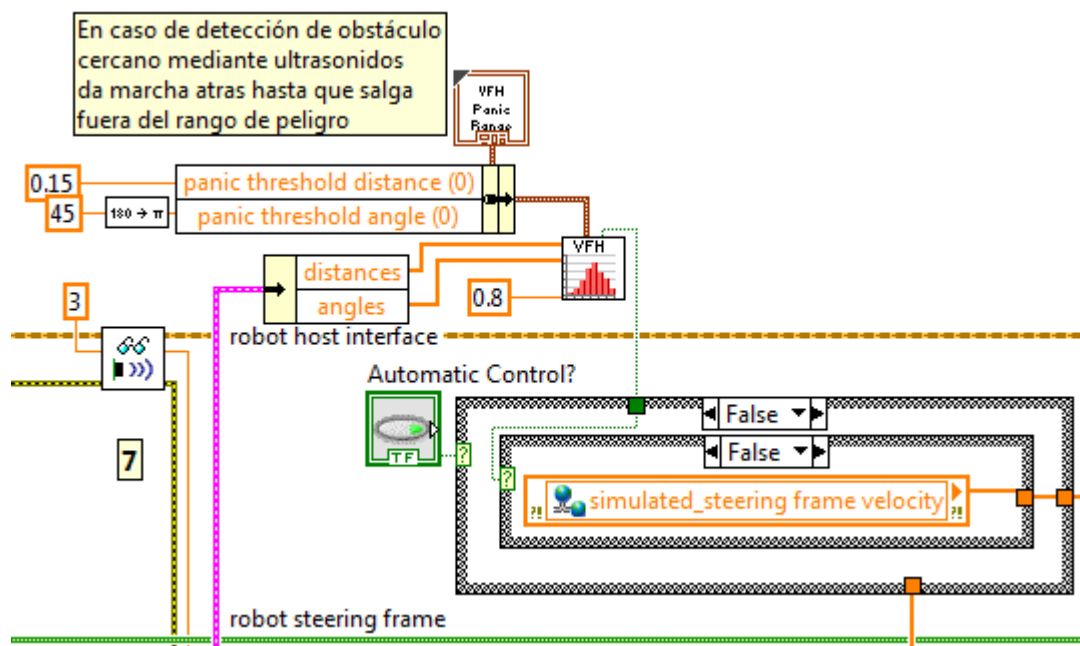


Figura 41: Control de la simulación y detección de obstáculos

5.3.2 CONTROL DE DaNI

Para el control de DaNI, como ya se ha comentado, gracias al planteamiento del proyecto, se crearon unas variables en el proyecto accesibles desde cualquier parte del mismo. Estas variables son comunicadas vía wifi a DaNI (Figura 42), donde se gestionan de la misma forma que en el simulador.

Se ha implementado en DaNI también un sistema anti colisión, que funciona de forma autónoma. La lectura del sensor de ultrasonidos gestionada por la FPGA, es transmitida al programa que se ejecuta en tiempo real en la sbRIO. Este programa monitoriza esta variable, y cuando detecta que se supera un rango de "pánico", se activa el sistema de protección. Este sistema de protección en un principio hacía que el vehículo se detuviera, pero esto obligaba al usuario a tener que recolocar el robot en una posición alejada del obstáculo para poder continuar el funcionamiento. Más tarde

esta gestión ha sido modificada de forma que el robot al detectar esa cercanía en lugar de pararse da marcha atrás un breve período de tiempo, dando tiempo al usuario a pararlo o cambiar de dirección si fuese necesario para sortear el obstáculo, y continúa con su normal funcionamiento.

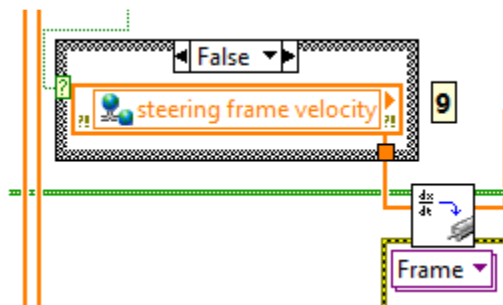


Figura 42: Lectura de las órdenes enviadas a DaNI

5.3.3 CONTROL DESDE ENOBIO

Desde Enobio, tras la realización de toda la estrategia de control, se comunican las órdenes pertinentes a un bucle que escribe en las variables destinadas a la comunicación desde este hardware (Figura 43). Será el programa principal quien acceda a estas variables y luego las comunique a DaNI, o a la simulación.

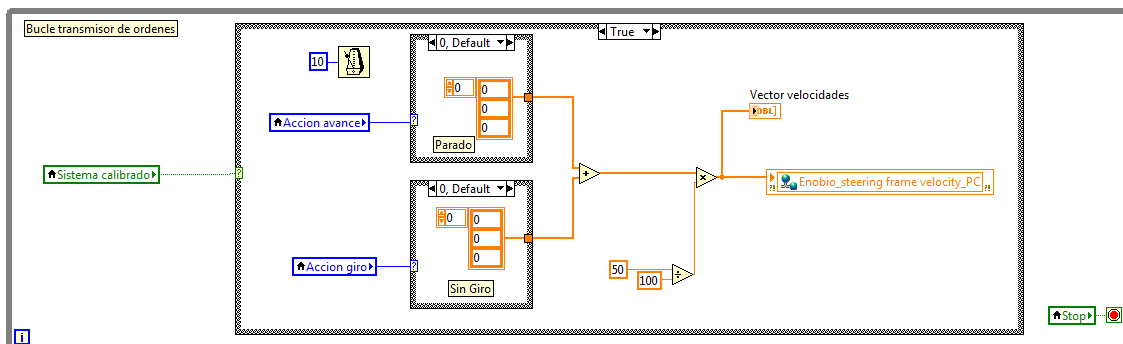


Figura 43: Escritura de variables desde Enobio

Hay que tener en cuenta que para poder realizar el control de una manera satisfactoria a través de Enobio, el usuario ha de estar a ser posible en una posición estable (sentado y con la espalda apoyada en un respaldo por ejemplo), con la pantalla del ordenador y el robot dentro de su campo visual en todo momento. Esta sería la situación natural si lo que se estuviera controlando fuese una silla motorizada y el conductor fuese sentado en ella.

5.3.4 CONTROL DESDE Wii Remote

Desde Wii Remote, el control de la velocidad y el movimiento de DaNI se comunica de la misma forma que desde Enobio, pero este control tiene una peculiaridad, y es que mientras que las velocidades que se obtienen realizando un control desde Enobio son constantes, estas dependen del grado de inclinación y giro del mando, lo que otorga al usuario un control mucho mejor del robot.

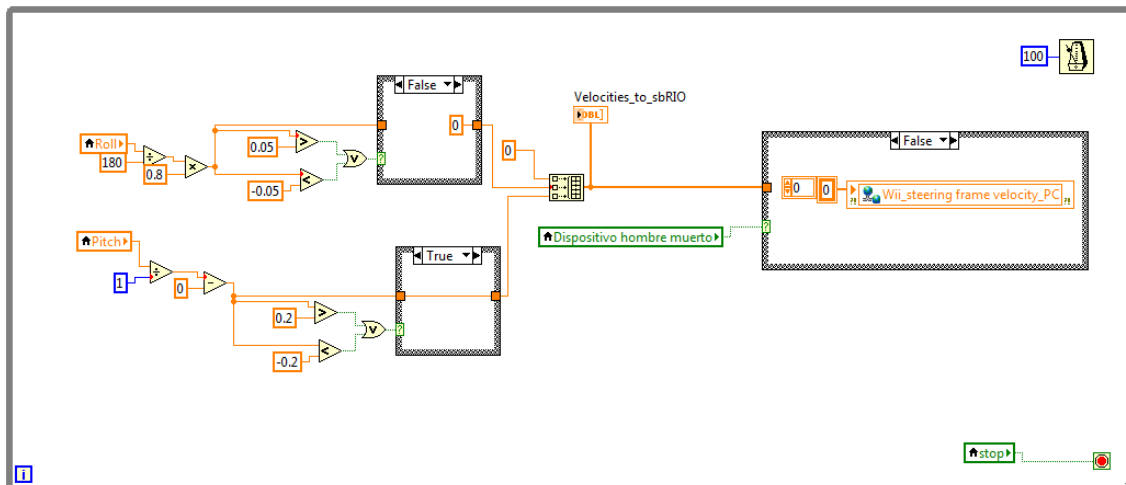


Figura 44: Escritura de variables desde Wii Remote

Para conseguir esta funcionalidad se tuvieron que comprobar los rangos de adquisición y el ruido de las señales adquiridas por los sensores del mando (Figura 44). Se establecieron unas pequeñas ventanas en las que se supone que el usuario no desea realizar movimiento alguno. Cuando esos umbrales son superados, la velocidad de DaNI es proporcional a la inclinación medida por los sensores.



Figura 45: Configuración de movimientos con Wii Remote

Para el control del robot mediante el mando de la Wii también se ha implementado un dispositivo de control de hombre muerto, de forma que el robot únicamente hará caso a las órdenes enviadas por el mando en caso de que se esté presionando el botón “B” del mando. En el momento en el que el botón no esté presionado se manda la orden de parar el robot. En la Figura 45 se puede observar la configuración de movimientos elegida para el control desde el mando Wii Remote.

5.3.5 CONTROL DESDE TECLADO/RATÓN

El control desde el teclado o ratón es la primera funcionalidad que se implementó (Figura 46), pero es sobre la que se han basado todas las demás. Se trata de un VI muy simple que escribe en las variables indicadas los valores calculados según una selección de controles y presionando unos botones. En este caso, la separación entre avance y giros no es efectiva ya que no pueden darse órdenes simultáneamente.

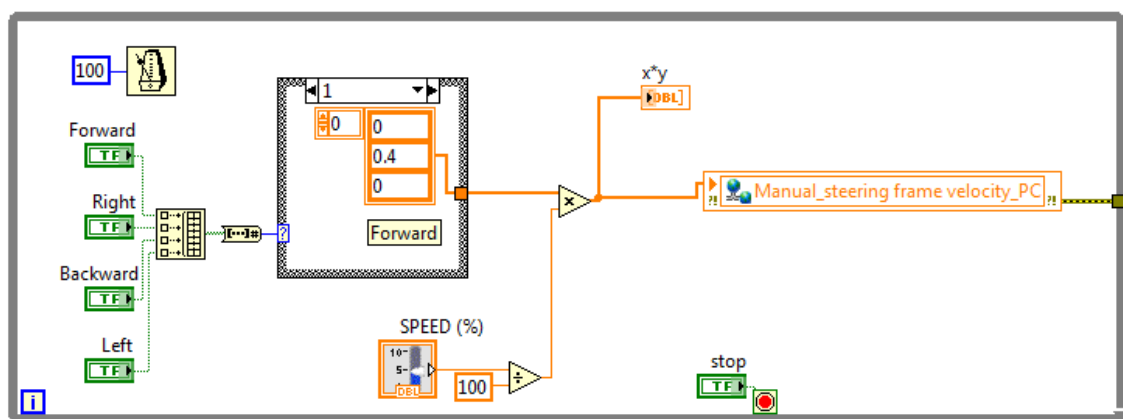


Figura 46: Escritura de variables desde teclado/ratón

A parte del control que se puede realizar presionando los botones con el ratón, se ha establecido las teclas “F1” y “F2” que corresponden a los botones “Forward” y “Backward” respectivamente, y las teclas “F11” y “F12” que corresponden a “Left” y “Right” (Figura 47), para poder controlar el robot directamente desde el teclado. Así como el control de velocidad que puede modificarse a través de las teclas “Page Up” y “Page Down” para incrementarlo o disminuirlo respectivamente. A continuación se puede observar una imagen de la ventana de configuración del teclado.

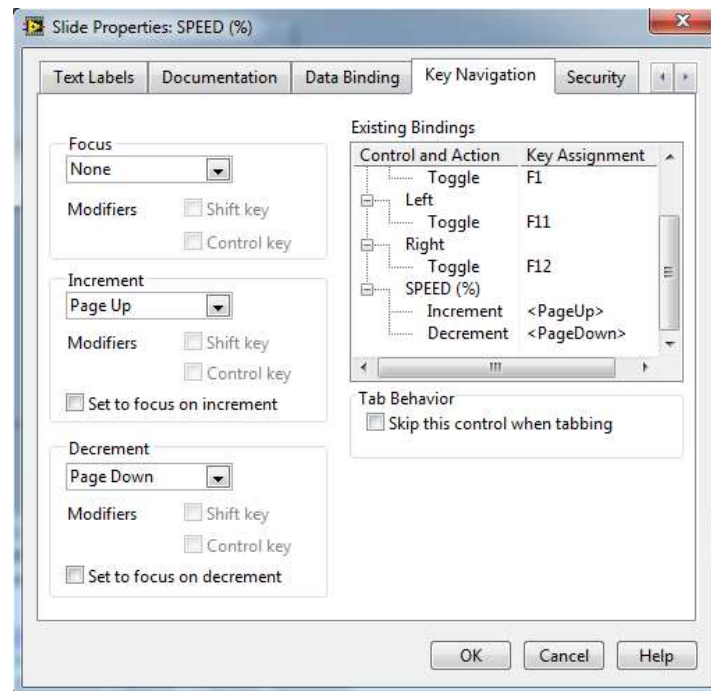


Figura 47: Ventana de configuración de la respuesta al teclado

5.4 RESULTADO Y DISCUSIÓN

Además de lo que se ha ido comentando a lo largo del proyecto, se reserva esta parte final para comentar ciertos aspectos del resultado del proyecto que no encajan en un capítulo concreto sino que forman parte del conjunto o bien no merecen un apartado para ellos mismos.

Apenas se ha hecho mención a la comunicación inalámbrica, y en el conjunto del proyecto nos encontramos con un enlace Wifi y otro Zigbee. El enlace wifi existente entre el ordenador y el robot se llevó a cabo a través de un aparato wifi ep-9501N que tuvo que ser configurado como punto de acceso, de esta manera establece una red inalámbrica a la que el ordenador puede conectarse, mientras que el aparato se conecta mediante cable a DaNI, y funciona con su propia batería, con una autonomía mayor que la de Enobio, por lo que no ha supuesto ninguna limitación en este sentido la utilización de este hardware. Una vez configurado el punto de acceso durante la primera parte del proyecto en que se trabajó en la conexión con DaNI no ha hecho falta volver a programarlo ni ha causado problema alguno ni en cuanto a autonomía (ya que es superior a la de DaNI), ni en cuanto a cobertura o interferencias. Se ha podido controlar el robot DaNI a más de 10 metros de distancia, más allá de lo cual no es conveniente aventurarse ya que el usuario puede no ser capaz de ver obstáculos que provoquen accidentes con el robot (como cables que podrían enredarse en sus engranajes).

En cuanto al otro enlace inalámbrico, el que conecta el casco de Enobio con la Enobio Box, se construye sobre las especificaciones de Zigbee. Teóricamente se

pueden alcanzar rangos de distancia de hasta 75 metros, sin embargo, durante el funcionamiento hemos comprobado que la transmisión de datos es bastante sensible a la distancia que separa ambos elementos, perdiendo algunos datos incluso cuando la distancia de separación no llega a ser de 20 metros. También cabe destacar un problema práctico al que hubo que enfrentarse en la demostración que se llevó a cabo en Madrid en IFEMA, y que se achaca al gran número de asistentes y electrónica concentrada en un mismo emplazamiento. En el congreso de NIDays España 2012, se encontró por primera vez que los dispositivos no se reconocían a no ser que la Enobio Box estuviera físicamente sobre el emisor del casco de Enobio. Sin embargo, el hardware Enobio tiene la posibilidad de ser reprogramado para emitir en cualquiera de los 16 canales establecidos en la norma IEEE 802.15.4. Y tras la reconfiguración del hardware la demo pudo ser presentada.

También cabe mencionar en este apartado los problemas de memoria que se han tenido con DaNI debido a la limitación de la sbRIO 9631. Ha dado muchos errores a la hora de desplegar los programas por esta causa, y en caso de haber podido utilizar alguno de los modelos más recientes de sbRIO podría haberse evitado, e incluso podría haberse planteado la implementación de más código de control automatizado.

6. CONCLUSIONES Y TRABAJOS FUTUROS

6.1 ANÁLISIS CRÍTICO Y CONCLUSIONES

El proyecto fue completado en plazos mucho más reducidos de los establecidos al principio. La duración estimada era de aproximadamente seis meses, pero por problemas burocráticos, de comunicación y vacacionales el hardware no se entregó en los plazos previstos. A pesar de esto, se han cumplido todas las especificaciones del proyecto y se han incluido muchas más funcionalidades que no estaban previstas en un principio por las empresas participantes.

Siguiendo los puntos que se concretaron al principio de esta memoria:

- Se ha completado de forma satisfactoria el acceso a la biblioteca proporcionada por la empresa Starlab para la adquisición y manejo de toda la información captada por el hardware Enobio. Y además en los últimos días de trabajo se ha conseguido alcanzar la máxima velocidad de adquisición y por lo tanto se puede concluir que se ha alcanzado el objetivo en su totalidad.
- Se han desarrollado las librerías necesarias para el acceso y configuración, no únicamente de las funciones básicas de Enobio, sino que se puede acceder a la configuración de distintas opciones como la selección de los distintos canales de transmisión.
- Toda la programación de DaNI y las modificaciones necesarias sobre programas previos se han podido realizar en sus límites establecidos de tiempo a pesar de los problemas por la falta de memoria en la placa que se han encontrado.
- Se ha implementado la programación de los movimientos básicos de avance y giro por separado. Se han añadido más funcionalidades según fuese el hardware de control uno u otro, como por ejemplo el sistema de detección de hombre muerto.
- Se ha llevado a cabo la realimentación mediante el uso del sensor de ultrasonidos para la creación del sistema anti colisión.
- Se ha mejorado el sistema anti colisión propuesto en un principio para evitar la intervención física del usuario en el caso de la detección de un obstáculo.
- Se ha implementado el control del robot desde el mando de la consola Wii, añadiendo además la funcionalidad de detección de hombre muerto.
- Se ha conseguido implementar el control de velocidad según la medida de los sensores inerciales que posee el mando Wii Remote.

- Se ha conseguido establecer una estrategia de control que ha permitido a diferentes usuarios ser capaces de seguir varios recorridos establecidos de forma aleatoria para comprobar si la detección y el control eran buenos.

Sin embargo y debido a la falta de tiempo por los retrasos que se han encontrado no han podido llevarse a cabo otros estudios complementarios que pretendían hacerse (aunque no entraban dentro de las especificaciones iniciales).

En general no solamente se ha conseguido llevar a cabo el proyecto, sino que pasó bajo la supervisión de responsables de National Instruments, y además de quedar sorprendidos de la calidad del código desarrollado, y encantados con la presentación, decidieron que el proyecto se presentase en una convención anual que se realiza una vez al año en cada país, y que este año en España se celebró en IFEMA el 22 de Marzo. Estuvieron varias empresas presentes mostrando los proyectos que realizaban, y también fueron presentadas varias demostraciones desarrolladas dentro de National Instruments, entre las que se encontraba este proyecto.

No sólo fue un éxito durante esta presentación, sino que suscitó también el interés de Greg Crouch, *Marketing and Executive Business Development*, actualmente liderando el *medical test and design Life Science Business Segment*. Tanto, que solicitó llevarse el proyecto a Austria, donde fue presentado en los NIDays el 28 de Marzo de 2012, en la *keynote*, por el propio Greg, recibiendo felicitaciones de nuevo por el proyecto realizado.

Por lo tanto se puede decir que se han alcanzado los objetivos del proyecto cumpliendo los límites establecidos en fechas. Ha sido una herramienta indispensable la integración con código externo que facilita LabVIEW, y la compatibilidad entre los diferentes programas y módulos que proporcionan los productos de National Instruments.

El apoyo del personal de Starlab cuando fue necesario hacer alguna consulta también fue de gran ayuda y respondieron sin tardanza. A pesar de que las complicaciones para el préstamo de Enobio pusieron en grave compromiso la ejecución del proyecto.

El resultado es un proyecto visualmente atractivo, fácil de comprender y con futuras aplicaciones si se sigue trabajando en la mejora de la adquisición y el tratamiento de las señales.

6.2 TRABAJOS FUTUROS

Debido a que es un proyecto realizado en colaboración con empresas, los trabajos futuros serán necesariamente a cargo de National Instruments.

Se ha propuesto ya dentro de la empresa el desarrollo de un conjunto de librerías para el acceso al hardware con propósito de comercializarlas como un *add-on*

de LabVIEW. Para lo que habría que cumplir unas exigencias que imponen a los desarrolladores de este tipo de bibliotecas.

Como otro posible trabajo futuro de este proyecto en su conjunto se plantea la posibilidad de estudiar otros tipos de filtros, sobre todo los wavelets, para mejorar la detección, o incluso generar una mejor estrategia de control, más intuitiva y con mejor respuesta a las acciones del usuario.

Podría realizarse un estudio más amplio sobre compatibilidad con distintos tipos de personas, diferentes edades, sexos, razas y otras condiciones de entorno para validar el sistema de forma más exhaustiva. Obtener estadísticas sobre usabilidad, respuesta, posibles problemas en otros entornos, etc.

Y por último, y dado que la conexión con DaNI se realiza a través de TCP/IP, podría modificarse el proyecto para controlar un robot conectado a internet en cualquier parte del mundo.

7. PRESUPUESTOS DEL PROYECTO

En este capítulo se detallan unos presupuestos para la realización del proyecto, calculados en función del valor del trabajo realizado, en el mercado empresarial.

Para el cálculo de todos estos presupuestos se han tenido las siguientes consideraciones:

- La duración del proyecto es de aproximadamente 6 meses, teniendo en cuenta que no se han incluido en el presupuesto los periodos (algunas semanas) en los que no se ha podido trabajar.
- La tasa de costes indirectos se considera de un 5% porque se han establecido unos costes bastante ajustados tanto en personal como en equipos y otros costes directos.
- Se estima que de los 6 meses trabajados, Dimas Gimeno Calvo ha trabajado a media jornada. Mientras que Alberto Jardón Huete ha dedicado momentos puntuales de supervisión y consejo dando como resultado una dedicación muy inferior.
- En la adquisición de equipos al conocerse de antemano las necesidades del proyecto, se han incluido los precios del paquete software con un descuento que otorga la empresa por la adquisición conjunta del software.
- Para los gastos de transporte se ha tomado como referencia la empresa TNT. Se han considerado los envíos entre Barcelona y Madrid, así como los envíos entre Madrid y Austria necesarios para llevar a cabo las presentaciones que se realizaron.

Los detalles de los presupuestos pueden encontrarse en las siguientes páginas.



UNIVERSIDAD CARLOS III DE MADRID

Escuela Politécnica Superior

PRESUPUESTO DE PROYECTO

1.- Autor: Dimas Gimeno Calvo

2.- Departamento: Electrónica y Automática

3.- Descripción del Proyecto:

- Título TELEOPERACIÓN DE UN ROBOT TERRESTRE MEDIANTE BIOPOTENCIALES
- Duración (meses) 6
- Tasa de costes Indirectos: 5%

4.- Presupuesto total del Proyecto (valores en):

Euros

5.- Desglose presupuestario (costes directos)

PERSONAL

Apellidos y nombre	Categoría	Dedicación (hombres mes) ^{a)}	Coste hombre mes	Coste (Euro)
Gimeno Calvo, Dimas	Ingeniero	3	2,694.39	8,083.17
Jardón Huete, Alberto	Ingeniero Senior	0.5	4,289.54	2,144.77
Total				10,227.94

^{a)} 1 Hombre mes = 131,25 horas. Máximo anual de dedicación de 12 hombres mes (1575 horas)

Máximo anual para PDI de la Universidad Carlos III de Madrid de 8,8 hombres mes (1.155 horas)

EQUIPOS

Descripción	Coste (Euro)	% Uso dedicado proyecto	Dedicación (meses)	Periodo de depreciación (meses)	Coste imputable ^{d)}
Enobio	300.00	100	3	36	25.00
LabVIEW Developer + FPGA + RT (descuentos)	11,383.46	50	5	60	474.31
LabVIEW Robotics	309.00	50	5	60	12.88
DaNI	2,300.00	100	4	36	255.56
Total					767.74

SUBCONTRATACIÓN DE TAREAS

Descripción	Empresa	Coste imputable
Total		0.00

OTROS COSTES DIRECTOS DEL PROYECTO^{e)}

Descripción	Empresa	Costes imputable
Gastos de oficina (ordenador, papel, toner, ...)		200.00
Gastos de transporte		60.00
desplazamientos		300.00
Total		560.00

^{e)} Este capítulo de gastos incluye todos los gastos no contemplados en los conceptos anteriores, por ejemplo: fungible, viajes y dietas, otros,...

6.- Resumen de costes

Presupuesto Costes Totales	Presupuesto Costes Totales
Personal	10,228
Amortización	768
Subcontratación de tareas	0
Costes de funcionamiento	560
Costes Indirectos	510
Total	12,133

Bibliografía

- [1] sillasderuedas.es, «Sillas de ruedas,» [En línea]. Available: <http://www.sillasderuedas.es/SPRINT-GTS-ELECTRICA-MEYRA>. [Último acceso: Mayo 2012].
- [2] J. C. L. Diez, Introducción a la biometría informática y análisis de huella dactilar como fuentes de autenticación en sistemas de seguridad, 2011.
- [3] g.tec, «<http://www.gtec.at/Products/Electrodes-and-Sensors/g.Electrodes-Specs-Features>,» Medical Engineering, Biosignal Electrodes, [En línea]. Available: <http://www.gtec.at>. [Último acceso: Marzo 2012].
- [4] Universidad de Oviedo, «Instrumentación biomédica,» [En línea]. Available: http://www.google.es/url?sa=t&rct=j&q=tina%20bioelectr%C3%B3nica&source=web&cd=1&ved=0CFQQFjAA&url=http%3A%2F%2Fwww.ate.uniovi.es%2F8695%2Fdocumentos%2Fclases%2520pdf%2Finstrumentacion%2520alumn.pdf&ei=guLQT4ToKIXQhAf7_-H-Cw&usg=AFQjCNEfNrWH9d8gzltHOu2rWq. [Último acceso: Mayo 2012].
- [5] L. T. Reilly RB, «Electrograms (ECG, EEG, EMG, EOG).,» *Technol Health Care.*, p. 15, 18 6 2010.
- [6] Ottobock, «<http://www.ottobockus.com>,» Prótesis y ortopedias, [En línea]. Available: http://www.ottobockus.com/cps/rde/xchg/ob_us_en/hs.xsl/6874.html. [Último acceso: Abril 2012].
- [7] U. d. M. Jorge Manrique, «Facultad de Ingeniería y Tecnología,» [En línea]. Available: <http://fit.um.edu.mx/jorgemp/clases/ingbiomedica/unidad2/IngBiomedicaUnidad2.pptx>. [Último acceso: Abril 2012].
- [8] Electrocardiografia.es, «Electrocardiografía.es,» [En línea]. Available: http://www.electrocardiografia.es/ondas_ecg.html. [Último acceso: 2012].
- [9] J.-P. Lachaux, «Increase of high-frequency (>150 hz) intracranial EEG activity during face perception in humans,» de *First International IEEE EMBS Conference*, Paris, France, 2003.
- [10] G. A. y. otros, «Analysis of breast cancer progression using principal component analysis and clustering,» *Journal of Biosciences*, p. 13, 2007.

- [11] S. C. y. otros, «Principal component analysis for reduction of ocular artefacts in event-related potentials of normal and dyslexic children,» *Clinical Neurophysiology*, vol. 115, p. 10, 2004.
- [12] S. Romero, «Reduction of EEG artifacts by ICA in different sleep stages,» de *25th Annual International Conference of the IEEE*, Barcelona, 2003.
- [13] Universidad Complutense de Madrid, «Docencia e investigación,» [En línea]. Available: <http://www.ucm.es/info/vivataca/anteriores/veintiuno/docencia.htm>. [Último acceso: Mayo 2012].
- [14] UNED, «Eye-Tracking Approaches,» Eye-Movement & Pupillometry Research Group, [En línea]. Available: http://www.uned.es/eyemovements-lab/links/tipos_movimientos.htm. [Último acceso: 2012].
- [15] y. o. Roberto Fraile, «Electrooculograma (EOG),» [En línea]. Available: <http://www.google.es/url?sa=t&rct=j&q=roberto%20fraile%20%C3%A1greda%20%C3%A1ngel%20soriano%20rodr%C3%ADguez%20%202C&source=web&cd=2&ved=0CFAQFjAB&url=http%3A%2F%2Fwww.seeic.org%2Fsociedad%2Fpremio%2Fdos.pdf&ei=l-fQT-jEDs6YhQey0qX8Cw&usg=AFQjCNHnAtIZw>. [Último acceso: Mayo 2012].
- [16] M. Reddy, «Complex wavelet transform driven de-noising method for an EOG signals,» de *India Conference (INDICON), 2011 Annual IEEE*, Hyderabad, India, 16-18 Dic. 2011.
- [17] Crocodile Clips Ltd, «Yenka.com,» Crocodile Clips Ltd, [En línea]. Available: <http://www.yenka.com/computing/>. [Último acceso: 2012].
- [18] Keep I.T. Easy, «Welcome to the Flowol Website,» [En línea]. Available: <http://www.flowol.com/Default.aspx>. [Último acceso: 2012].
- [19] ABACOM, «ABACOM Electronics-Software,» [En línea]. Available: <http://www.abacom-online.de/uk/html/profilab-expert.html>. [Último acceso: 2012].
- [20] National Instruments, «NI LabVIEW,» [En línea]. Available: <http://www.ni.com/labview/esa/>. [Último acceso: 2012].
- [21] Emotiv, «Emotiv. You think, therefore, you can,» [En línea]. Available: <http://www.emotiv.com/>. [Último acceso: Marzo 2012].
- [22] Neuro Sky, «Brain wave sensors for every body,» [En línea]. Available: <http://www.neurosky.com/Products/MindWave.aspx>. [Último acceso: Marzo 2012].

BIBLIOGRAFÍA

- [23] Pitsco Education, «Pitsco, NI collaborate to create DaNI 2.0,» [En línea]. Available: <http://shop.pitsco.com/about/news/item.aspx?art=5000&bhcp=1>. [Último acceso: 2012].
- [24] UNE-EN 61326-1:2006, «Material eléctrico para medida, control y uso en laboratorio. Requisitos de compatibilidad electromagnética (CEM).,» UNE, 31-10-2006.
- [25] UNE-EN 55011:2008/A2:2008, «Límites y métodos de medida de las características relativas a las perturbaciones radioeléctricas de los aparatos industriales, científicos y médicos (ICM) que producen energía en radiofrecuencia.,» UNE, 02-04-2008.
- [26] UNE-EN 61000-4-2:1997, «Compatibilidad electromagnética (CEM). Parte 4: Técnicas de ensayo y de medida. Sección 2: Ensayos de inmunidad a las descargas electrostáticas. Norma básica de CEM.,» UNE, 09-06-1997.
- [27] UNE-EN 61000-4-3:2007, «Compatibilidad electromagnética (CEM). Parte 4-3: Técnicas de ensayo y de medida. Ensayos de inmunidad a los campos electromagnéticos, radiados y de radiofrecuencia.,» UNE, 23-03-2011.
- [28] National Instruments, «National Instruments,» [En línea]. Available: www.ni.com. [Último acceso: Mayo 2012].
- [29] CodePLEX, «Managed Library for Nintendo's Wiimote,» 2009. [En línea]. Available: <http://wiimotelib.codeplex.com/>. [Último acceso: 2011].
- [30] Sshearman, «LabVIEW interface to a Wii Remote ("Wiimote"),» 2008. [En línea]. Available: <https://decibel.ni.com/content/docs/DOC-1353>. [Último acceso: 2011].
- [31] R. C. y. otros, *Electrocardiograma*, Málaga: Servicio de Cardiología del Hospital Virgen de la Victoria (Málaga)..
- [32] National Instruments Corporation, LabVIEW Core 2 Course Manual, Hungary: National Instruments, 2010.
- [33] National Instruments Corporation, LabVIEW Core 1 Course Manual, Hungary: National Instruments, 2011.
- [34] National Instruments Corporatin, LabVIEW Real-Time Course Manual, Hungary: National Instruments, 2009.
- [35] National Instruments Corporation, LabVIEW FPGA Course Manual, Hungary: National Instruments, 2009.

Figuras

Figura 1: Silla de ruedas motorizada[1].....	1
Figura 2: Diagrama de Gantt.....	5
Figura 3: Posiciones de los electrodos para estudios EEG en personas[4]	9
Figura 4: Interferencias externas[7].....	12
Figura 5: Acoplo capacitivo[7].....	13
Figura 6: Bucle de interferencia y disposición de los cables [7].....	13
Figura 7: Filtros paso bajo, alto y banda	15
Figura 8: El ojo humano[13].....	16
Figura 9: Registro de movimientos oculares[15].....	18
Figura 10: Ejemplo de programación gráfica en Profilab Expert[19]	21
Figura 11: Programación en LabVIEW.....	22
Figura 12: Dispositivo Mind Wave de NeuroSky[22].....	23
Figura 13: Mando Wii	24
Figura 14: Esquema de control del proyecto.....	25
Figura 15: Esquema de conectividad del proyecto.....	26
Figura 16: Software Enobio	28
Figura 17: Diagrama de montaje de DaNI[28]	29
Figura 18: Dibujo de sbRIO 9631[28]	30
Figura 19: Panel frontal y diagrama de bloques de un VI.....	32
Figura 20: LabVIEW Robotics[28].....	32
Figura 21: LabVIEW FPGA.....	34
Figura 22: Ventana de reporte de DIAdem.....	35
Figura 23: Esquema del enlace LabVIEW - Enobio	38
Figura 24: Función de transferencia de datos del wrapper	38
Figura 25: SubVI de adquisición de datos	39
Figura 26: Ventana de proyecto, simulación integrada.....	40
Figura 27: DaNI simulado, control manual	41
Figura 28: Ventana de proyecto, Wii Remote integrado	42
Figura 29: Esquema del enlace LabVIEW - Wii Remote	43
Figura 30: Diagrama de bloques, enlace RT - FPGA.....	44
Figura 31: Generación de PWM en la FPGA para el control de un servo.....	45
Figura 32: Panel frontal de comunicación con Enobio.....	48
Figura 33: Transmisión de datos entre Enobio y LabVIEW.....	48
Figura 34: Acceso a biblioteca, adquisición, filtrado y puesta en cola de los datos adquiridos... ..	49
Figura 35: Tratamiento de los datos obtenidos.....	49
Figura 36: Creación y registro de eventos en el VI principal	50
Figura 37: Funciones de registro y envío de eventos en el wrapper	50
Figura 38: Máquinas de estados	53
Figura 39: SubVIs de calibración, avance y giro	55
Figura 40: Variables publicadas en el proyecto	56
Figura 41: Control de la simulación y detección de obstáculos	57
Figura 42: Lectura de las órdenes enviadas a DaNI	58
Figura 43: Escritura de variables desde Enobio.....	58
Figura 44: Escritura de variables desde Wii Remote.....	59
Figura 45: Configuración de movimientos con Wii Remote.....	59
Figura 46: Escritura de variables desde teclado/ratón.....	60
Figura 47: Ventana de configuración de la respuesta al teclado	61

Tablas

<i>Tabla 1: Magnitudes de los electrogramas [7]</i>	10
<i>Tabla 2: Especificaciones técnicas de Enobio</i>	27
<i>Tabla 3: Características del sensor ultrasónico</i>	29

Gráficos

<i>Gráfico 1: Esquema de la señal ECG [8]</i>	11
<i>Gráfico 2: Lectura de canales sin filtrar en DIAdem</i>	52

Glosario de términos

A

API, 3, 28, 37

Aplication Programming Interface

B

BCI, 1, 6, 10

Brain-Computer Interface

C

CMRR, 27

Common Mode Rejection Ratio.

CWT, 19

Complex Wavelet Transform.

D

DLL, 37

Dynamic-Link Library.

E

ECG, 10, 14, 19, 26, 51

Electrocardiograma.

EEG, 1, 3, 9, 14, 19, 26, 51

Electroencefalograma.

Electrodo DRL, 27

Driven-Right-Leg circuit: Se trata de una conexión muy común en las medidas de biopotencial.

Típicamente en las medidas de electrocardiogramas, cuando el paciente se encontraba tumbado, se conectaba un electrodo a la pierna derecha y a masa para disminuir el ruido de las señales medidas. Ahora se llama así también a los electrodos de referencia en dispositivos portátiles.

EMG, 10, 12, 14

Electromiograma: Lectura de biopotenciales registrados en un músculo.

Enfermedad de Best, 19

La enfermedad de Best, también conocida como distrofia macular viteliforme, se caracteriza por una pérdida de la visión central similar a la observada en otras formas de degeneración macular. Normalmente, la enfermedad se diagnostica en la infancia.

Enfermedad de Stargardt, 19

La enfermedad de Stargardt es una enfermedad genética. Se caracteriza por la muerte prematura de los conos, células responsables de la recepción y transmisión de la luz al nervio óptico.

EOG, 1, 3, 9, 12, 14, 26, 51

Electrooculograma.

Epitelio pigmentario retinal, 19

El epitelio pigmentario retinal o EPR es una capa de células pigmentadas que aparece en el exterior de la retina que nutre sus células visuales.

ERG, 16

Electroretinograma.

ERP, 11

Event-Related Potential (ver PEV).

F

FPGA, 2, 4, 30, 33, 44, 57, 69

Field-Programmable Gate Array.

G

GPS, 23, 35

Global Positioning System.

I

ICA, 15

Independent Component Analysis.

L

LabVIEW, 1, 20, 28, 31, 37, 41, 47, 64, 69

Laboratory Virtual Instruments Engineering Workbench.

P

PCA, 15, 16

Principal Component Analysis.

PCB, 30

Printed Circuit Board.

PEV, 16

Potencial Evocado Visual: Se trata de un cierto tipo de estudio que trata de medir la reacción.

Prótesis mioeléctrica, 10

Prótesis que responde a los impulsos eléctricos enviados a un músculo (aunque este esté atrofiado, o no esté).

GLOSARIO DE TÉRMINOS

R

REM, 17, 20

Rapid Eyes Movement: Se llama así a una fase del sueño de las personas que se caracteriza precisamente por esos movimientos rápidos que hacen los ojos.

RT, 33, 43, 69

Real Time.

S

sbRIO, 2, 4, 23, 29, 30, 43, 57, 62

Single Board Reconfigurable Inputs and Outputs.

V

VI, 2, 31, 43, 48

Virtual Instrument, es archivo de programación de LabVIEW.

W

Wrapper, 4, 37, 48

Es en términos de computación una capa software capaz de acceder a una capa separada de manera que el usuario pueda interactuar con la última a través de unas ordenes específicas y programadas de manera independiente en el wrapper limitando así el acceso, u obligando a acceder de cierta forma específica.